

EXE

JULY 1990

VOL 5

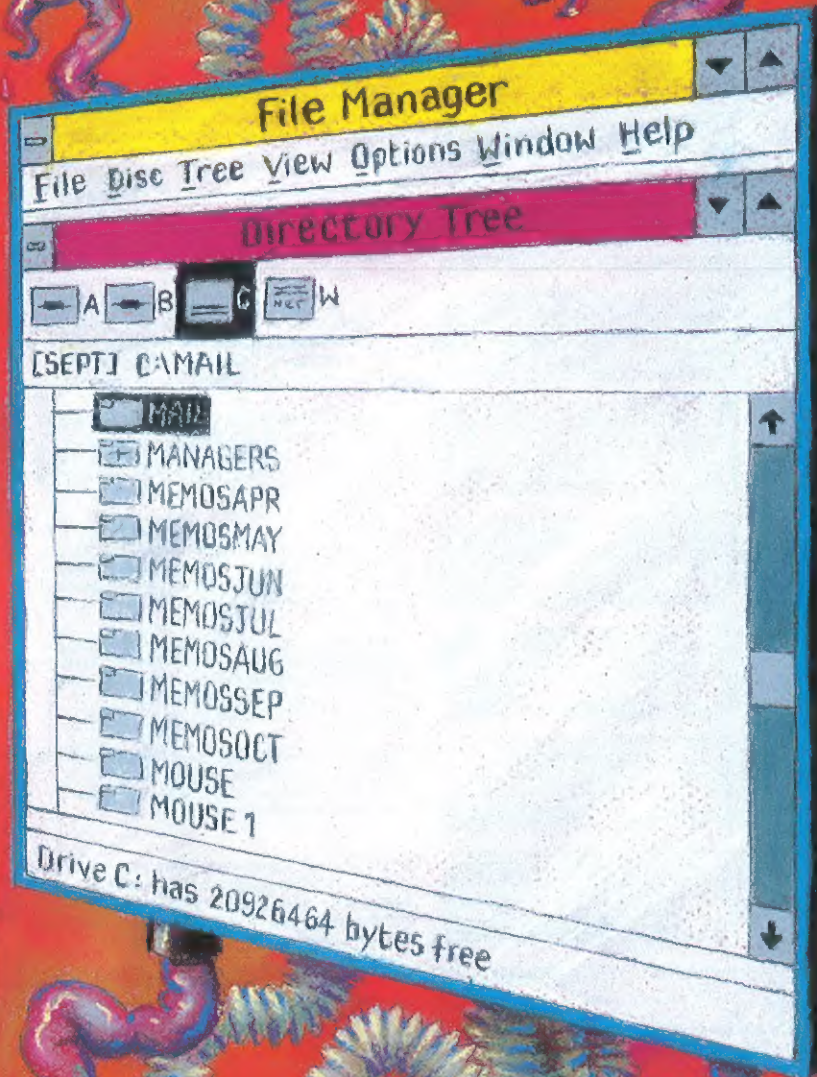
ISSUE 2

SCOTTISH OFFICE

2 JUL 1990

£3.20

The Software Developers' Magazine



Inside Windows 3. We have a programmer's eye view from behind its glitzy façade.

Here is Borland's latest Turbo language: C++

No man is an island, even when surrounded by C. We start a new opinion column.

Some OOP building blocks for Turbo Pascal users.

Sequence, Selection, Iteration and...? The fourth kind of programming structure takes a bow.

To 'C' PROGRAMMERS AT ALL LEVELS

"The best path for complete 'C' training begins at QA".

'C' has become the dominant language in computer programming and provides access to the software industries major application program interfaces.

'C' is probably now, or shortly will be, your way forward in program development – and QA can really help you along that path.

Only QA's extensive 'C' curriculum totally supports the training needs of all 'C' programmers, from beginner through to highly experienced professional. Each course can stand alone or can be entry point to reach any level of skill or knowledge you may set yourself.

In common with all QA courses, our instructors are experienced professionals themselves, our training materials are perfected for efficient learning, and all our own skill and energy is channelled toward helping you get the maximum benefit from your training.

Send for more details today or call Samantha Trinder on 0285 655888.

BEGINNERS START HERE

OTHERS START HERE...

...OR ANYWHERE HERE...

...INCLUDING C++

'C' COURSE OPTIONS

C PRIMER 2 Days £340

HLL concepts via C • Overview of C programming environment • Developing simple C programs • Data typing, block structure, scope concepts • Introducing C functions and modular programming

C PROGRAMMING 4 Days £720

C programming structure and style • Data types and variables • Flow of control in C programs • Pointers and address manipulation • Portability and efficiency considerations • Debugging C programs • Working with larger C programs

C TECHNIQUES FOR OS/2 & P.M. 1 Day £180

Using C in an OS/2 environment • PC and OS/2 extensions to C • Mixed Models of compilation • Advanced C techniques for OS/2 • The OS/2 API C interface • Dynamic link library concepts • OS/2 development tools overview

C TECHNIQUES FOR DOS 1 Day £180

Implications of using C under DOS • PC and DOS extensions to C • Models of compilation and segmentation • Interfacing to DOS and BIOS from C • Linking C to other languages • DOS development tools overview

C ADVANCED SOFTWARE DEVELOPMENT 4 Days £760

A lightning tour of C • Advanced pointer techniques • 'Defensive' programming techniques in C • Advanced C concepts • A closer look at C's library routines • Recent ANSI and SAA C developments • C development tools overview

C STANDARDS 1 Day £220

C standards – aims, goals and benefits • Defining practical and realistic standards • Naming and layout standards • C program organisation standards • Defensive C programming • Portability, maintainability and efficiency

C++ PROGRAMMING 4 Days £780

Object oriented programming concepts • OOP and C++ • Concept of classes • Class hierarchies, derivation and inheritance • Constructors and Destructors • Application examples

OBJECT ORIENTED DESIGN USING C++ 3 Days £620

OOP and OOD: the need • OOD: fundamental concepts • Identifying objects and relationships • Object properties • Stages in object oriented design • C++ design guidelines • Development team roles • Tooling up for OOD



Please complete and mail to QA Training, Cecily Hill Castle, Cirencester, Glos GL7 2EF.

☐ Tick here for full details on all our 'C' courses.

Send me details on (tick as appropriate):

- ☐ C Primer ☐ C Programming ☐ C Techniques for OS/2 & PM
☐ C Techniques for DOS ☐ C Advanced Software Development
☐ C++ ☐ OS/2 Courses ☐ Windows Courses ☐ Support Courses

CIRCLE NO. 060

NAME

TITLE

COMPANY

ADDRESS

TEL. NO.

QA
TRAINING

Editor: Willie Watts
Editorial Assistant: Wendie Leek
Display Advertising Manager: Sandra Inniss-Palmer
Display Sales Executive: Ed Butcher
Group Recruitment Sales Manager: Jane Lyons
Recruitment Sales Executive: Jeremy Cox
Reader Services & Promotions: Helena Adams
Design & Layout: Katerina Adams
 Mark English
Office Administrator: Carla Pearce

COVER ILLUSTRATION: LYNNE CHAPMAN

General

EXE Magazine is independent and not affiliated to any vendor of hardware, software or services. It is published by Process Communications Ltd, 10 Barley Mow Passage, Chiswick, London W4 4PH.

Tel: (Advertising/Editorial/Production) 081 994 6477
 (Subscriptions) 044282 4501

Facsimile: 081 994 6477 Telex: 8811418 SPACES G
 ISSN: 0268-6872

Subscriptions

EXE Magazine is a monthly journal for software developers. It is available only by subscription, at a cost of £35 per annum (11 issues) in the UK (see insert card for academic and overseas rate). A subscription implies that this journal will be sent to the subscriber until one of the three expires - AG MacDonell. The magazine is published around the 26th of the month preceding the cover date. There is no January issue. To subscribe, please call 044282 4501, or write to The Subscriptions Manager, EXE Magazine, 10 Barley Mow Passage, Chiswick, London W4 4PH. We can invoice your company or take ACCESS and VISA cards. In case of a query about your subscription, please call 044282 4501.

Back issues are available at a cost of £3.50 each - call our Chiswick office (081 994 6477) for a list of issues still in print. We also offer a range of magazine binders, disk holders etc - please call for details.

Editorial

Editorial enquiries should be addressed to The Editor, EXE Magazine, 10 Barley Mow Passage, Chiswick, London W4 4PH. We welcome letters, opinions, suggestions and articles from our readers. If you are interested in contributing articles, please write to this office for a copy of our Contributors' Guide.

Information contained in EXE is believed to be correct. If errors are found, we will endeavour to publish a clarification in the next available issue.

From time to time, we offer to copy the PC program code described in an article onto diskette for our readers. In this case, please send a blank, formatted disk with a self-addressed, prepaid mailer to the editorial address given above. We can copy both 5.25" and 3.5" disks.

Advertising

If you are interested in advertising in this magazine, please write to the Display Advertising Manager/Recruitment Sales Manager (as appropriate), EXE Magazine, 10 Barley Mow Passage, Chiswick, London W4 4PH, or call 081 994 6477 for details of our advertising rates.

Pronunciation

The name of EXE Magazine is pronounced to rhyme with 'not sexy magazine'.

Copyright

Material published in EXE Magazine is copyright © Process Communications Ltd. Articles (or parts of articles) may not be copied, distributed or republished without written permission from the publishers.

INSIDE WINDOWS 3

What do the innards of Microsoft's latest pseudo OS look like? Peter Morris brings us up to date with the world of Windows.

12

RED IS THE COLOUR OF TURBO C++

Turbo C++ has descended among us. Willie Watts has been taking a look at the latest compiler from the Borland stable.

18

NOT ENTIRELY UNBIASED

Steve Teale is an expert on MS-DOS C++ compilers. He is the ideal candidate for a Turbo C++ review... except he works for Zortech.

24

PROGRAM STRUCTURES OF THE FOURTH KIND

GO TO sorely missed. Richard Pickard doesn't want to go back to the bad old days - he has a better idea.

27

OS/2 DYNAMIC LINK LIBRARIES

Graeme Burton takes us, step by step, through the process of building our own DLLs for OS/2.

36

TURBO PASCAL OOP TEXT WINDOWS

At last, a practical example of a class library. David Bolton presents his Turbo Pascal code.

42

SOAPBOX

Jules May, we hear, doesn't use C. Neither can he understand why anybody else does.

2

NEWS

What happened when Bjarne Stroustrup came over here, a class library for C++ programmers, a dBASE compiler for UNIX users.

4

LETTERS

A sharp rebuff from an APL specialist; one of our bugs revealed plus some of Microsoft's.

10

THE THIRD SIDE

Coral 66 is definitely a Real Time pioneer. It's past its sell-by date now, but Jim Cooling still loves it.

52

CODE PAGE

This month: Robert Schifreen presents a simple debugging application.

60

CROSSWORD

Our regular puzzle - with solutions to earlier months' problems.

62

UNIX REGULAR

Peter Collinson describes a surprising feature in the handling of command line arguments.

64

BOOKS

We look at two books about networking.

68

CAREER NEWS

A new regular feature. Included this month: a look at the gains and drawbacks of ethical jobs.

76

STOB

Verity has a family! Parity is as cynical as her sister.

80

Jules May doesn't use C

Why is it that a language which disobeys all the established rules of language design is used for nearly all production programming? Jules May explains why he will never use C again.

Programs are among the largest and most complex things that are built by people; the activity of producing a program is not easy at the best of times. In the process of doing something so hard, the tools one uses figure very highly in the success or failure of the project, and paramount among those tools is the language used to create the program.

It seems the national sport of programmers is to argue about which language is the best to program in. This is a game I totally fail to understand - the idea that there is a best language for all possible applications, or even for a single production program, seems entirely fatuous to me. However, there is such a thing as a worst language, a language which incorporates all the features of the oldest and most unpleasant languages ever built, with none of their advantages, and a few new wrinkles thrown in for good measure - such a language is C.

What do I mean by a bad language? It seems to me that a production program must work. No matter how ambitious the design, no matter what tricks have been used, no matter how good the user interface, if a program fails in use, it is badly made. Such failures need not be debilitating to make the program unusable; if a user spends five minutes thinking of a work-around and forgets his place in his job, that bug has cost perhaps half an hour of real time.

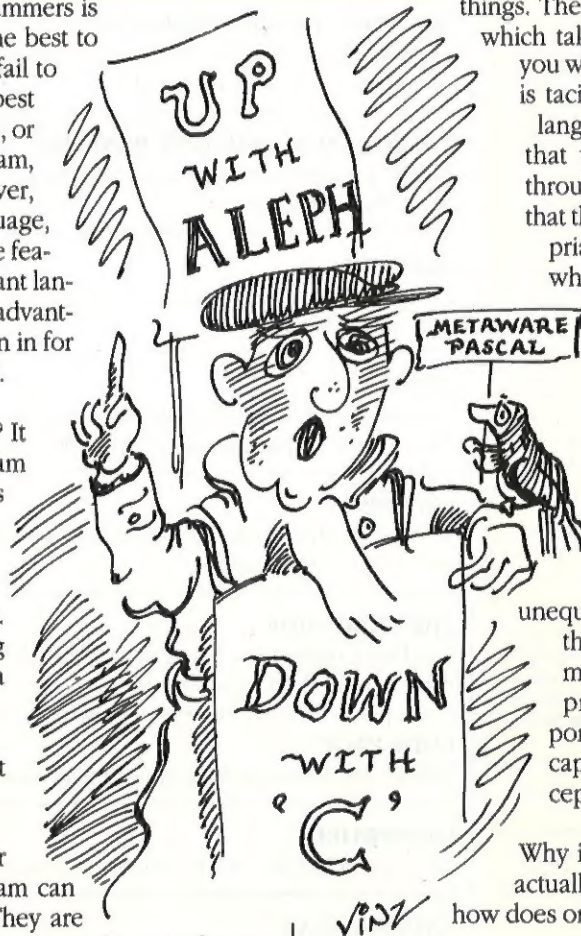
There are people who have built their careers on explaining why a program can never be made completely clean. They are wrong; it is possible to write clean programs, but not if you're writing in C. The C supporters claim that one is allowed access to the most secret parts of the machine. I agree, and I think this is a laudable feature in a language; my complaint is that one doesn't have the option not to. One can't think in terms of metaphors: one has to allocate memory explicitly, pass around its addresses explicitly, then deallocate it when one is sure one has finished with it. Every C programmer knows what happens when memory is deallocated prematurely; the resultant errors propagate around the computer until even the operating system is destroyed. How can anyone defend the concept of a compilable program being able to crash the operating system?

Computers were created to do boring, repetitive and precise jobs that are beneath human dignity. Jobs like turning references to reals into references to doubles. Jobs like checking that a variable passed by reference is not received by value. Jobs like deciding whether memory is in use or not. Jobs like making sure that a program stays decently within the bounds that the operating system gives it. C, you will notice, does none of these things.

The existence of programs such as *lint*, which takes a compilable program and tells you what you have done that is dangerous, is tacitly acknowledging the faults of the language. It is intrinsic to the language that you can pass almost any rubbish through the stack; the assumption is made that the programmer is able to take appropriate care at the receiving end, even when the receiver is in a different file from the sender - the actual file not being obvious from any piece of code, but with perhaps 50 to choose from on a significant project.

As if this were not enough, C programmers claim with perfectly straight faces that a language which is advertised as providing unequalled control over the deep levels of the machine (apparently to the detriment of all civilised behaviour), also produces code which is inherently portable. I do not understand a mind capable of believing both of these concepts at the same time.

Why is C so popular? Why would anyone actually want to write a program in it, and how does one get such a program to work?



Jules May is a freelance programmer specialising in graphics and HCI. He also runs Jules Computer Ltd, a company marketing a 3D graphics sub-system which runs on a range of machines. In 10 years of programming, he has only produced one bug. He can be contacted on 0707 44185.

We are looking for contributors to this new column. If you have an opinion or idea that you would like to air here, please contact The Editor at the address given on page 1.

THE C LANGUAGE

Microsoft C V6 is a complete rewrite with improved optimisation and a new Programmer's Workbench. High C V1.6 has been considerably improved, with better Microsoft C compatibility, and new documentation.

C COMPILERS

Avocet C	PC-DOS	£210
Aztec C86 Professional	MS-DOS	£120
Aztec C86 Developer	MS-DOS	£175
Aztec C86 Commercial	MS-DOS	£290
CI C86 Plus v1.3	MS-DOS	£290
ECO-C88 V4.0	MS-DOS	£ 70
HIGH C (Metaware) v1.6	MS-DOS	£420
HIGH C 386 v1.6	PL386&MS-DOS	£635
Hi-Tech C	MS-DOS	£135
Lattice C v6	OS/2&MS-DOS	£175
Let's C v4	MS-DOS	£ 60
Microsoft C v6	OS/2&PC-DOS	£269
Microsoft QuickC v2	PC-DOS	£ 60
M'soft QuickC/QuickAsm	PC-DOS	£105
MIX C	MS-DOS	£ 20
MIX Power C & lib s'ce	PC-DOS	£ 29
Turbo C v2	PC-DOS	£ 79
Turbo C Professional v2	MS-DOS	£160
Turbo C++	PC-DOS	£140
Topspeed C for OS/2	OS/2	£335
Topspeed C (std)	PC-DOS	£135
Topspeed C Professional	PC-DOS	£270
Watcom Express C v7	PC-DOS	£ 65
Watcom v7 opt comp + tlpC-DOS	PC-DOS	£250
Watcom C/386	PL386&MS-DOS	£625
Zortech C v3	PC-DOS	£ 48
Zortech C++ v2	PC-DOS	£120
Z'tech C++ v2 Developer	PC-DOS	£270
Aztec C86 Developer	CP/M-86	£245
Aztec C86 Personal	CP/M-86	£140
Hi-Tech C	CP/M-86	£150
Lattice C	CDOS	£395
HIGH C (Metaware)	Flex OS 286	£420
Avocet C	CP/M-80	£210
Aztec C Personal 1.06D	CP/M-80	£120
Aztec Commercial 1.06D	CP/M-80	£200
Eco-C w. Lib s'ce	Z80 + CP/M-80	£ 60
Hi-Tech C	Z80 + CP/M-80	£ 99
Mix C	Z80 + CP/M-80	£ 35
Toolworks C/80 v3.1	CP/M-80	£ 45
Aztec C65 V1.05	Apple DOS	£200
Aztec C65 V3.20	Apple PRO-DOS	£245
Aztec C68K MPW C	MACINTOSH	£110
Aztec C68K C	MACINTOSH	£ 75
Aztec C68K/AM Developer	AMIGA	£175
Aztec C68K/AM Prof'snl	AMIGA	£120
Aztec C68K/ST Developer	ATARI	£175
Aztec C68K/ST Prof'snl	ATARI	£120
Hi-Tech C	ATARI	£ 99
Lattice C v3.04	ATARI	£ 85
Mark Williams C v2	ATARI	£110
Laser C	ATARI	£135
Prospero C	ATARI	£105

PROGRAMMING TOOLS

AdaCompilers	Algol Compilers
Assemblers & Libs	AWK
Basic Compilers	Basic Interpreters
Basic Utilities	Basic Libraries
BCPL Compilers	C Compilers
C Interpreters	C Libraries
C Utilities	Cobol Compilers
Comms.Libraries	Cross Assemblers
Database Libs.	Debuggers
Dis-assemblers	Editors
Engineers Libs.	Expert Systems
Fortran	Fortran Compilers
Fortran Libraries	Graphics Libraries
Icon	Linkers/Locaters
Lisp	Modula-2
Nial Interpreters	OPS 5
Pascal Compilers	Pascal Libraries
Prolog	Rexx
Risc	Screen Libraries
Smalltalk	Snobol

We stock many items for which there is no space in these advertisements.

CROSSASSEMBLERS

We supply cross-assemblers by Avocet, 2500AD, IAR Systems and Pecan hosted on MS-DOS, CP/M-86 and CP/M-80 with over 30 target processors. In total over 200 products with no space to list them here. We hold some stock but you should allow 10-14 days for delivery. Please call for information or advice.

C CROSS COMPILERS

We supply 2500AD, Avocet, Aztec, Lattice, IAR and Hi-Tech Cross Compilers hosted on MS-DOS and targeted on Z80, 6502, 6801, 68HC11, 6301, 6809, 7811, 8051, 8096, 68000 & 68020. Please call for information or advice.

C INTERPRETERS

Instant C/16M supports large programs on 286/386 machines.

C-terp V3.0X	PC-DOS	£165
Interactive C	PC-DOS	£195
Introducing C	PC-DOS	£ 85
Living C Plus	PC-DOS	£135
RUN/C Professional	MS-DOS	£110

DISK COPYING SERVICE

We can copy files to and from 600 disk formats including CP/M, CP/M-86, MS-DOS, PC-DOS, ISIS, APPLE, SIRIUS, BBC, TORCH, APRICOT, HP-150, TRSDOS, DEC RT-11, IBM BEF, ATARI ST, AMSTRAD, MACINTOSH. Our charge is £10.00 + disk + VAT with discounts on small quantities and disks are normally despatched within 24hrs of receipt. For more information call us.

CLIBRARIES

DATABASE

Btrieve	OS/2	£400
Btrieve v5	MS-DOS	£170
Btrieve/Network	MS-DOS	£400
XQL	OS/2, PC-DOS	£540
Xtrieve Plus	OS/2, PC-DOS	£400
C/Database T'chest s'ce	PC-DOS	£ 39
CBTREE	(source any C)	£135
C-Index/Plus	(source any C)	£175
C-ISAM (L,MS)	MS-DOS	£220
C-tree	(source any C)	£230
Essential Btree (s'ce)	PC-DOS	£120
R-tree	MS-DOS	£170
db-VISTA III s'gle user	PC-DOS	£575
db-Query v2 s'gle user	PC-DOS	£575
Lattice dBCIII (L,MS)	MS-DOS	£165
Lattice dBCIII+	OS/2&MS-DOS	£385
SftFocis Btree&Isam (s'ce any C)	£ 80	
Virtual Memory File Man.	PC-DOS	£ 80
Zortech Btree/ISAM	PC-DOS	£ 75

GRAPHICS

Enhanced Graphics Tlkt	PC-DOS	£170
Essential Graphics v3	PC-DOS	£235
GFX Graphics	PC-DOS	£ 75
GraphicC v5 (CI, L, DS, MS)	PC-DOS	£290
Graphics Menu/META	PC-DOS	£135
GSS Kernel system	OS2, PC-DOS	£350
GSS Graph Dev T'kit OS2	PC-DOS	£350
HALO v3 (MSC5, LAT, TC)	PC-DOS	£230
HGraph (MSC)	PC-DOS	£ 80
Ingraf (MSC)	PC-DOS	£175
MetaWINDOW (CI, L, MS, T)	PC-DOS	£145
MetaWINDOW/Plus	PC-DOS	£190
MetaWINDOW/Premium	PC-DOS	£315
TurboWINDOW/C (Turbo)	PC-DOS	£ 79
QuickWINDOW/C (QC)	PC-DOS	£ 79

SCREEN & WINDOWS

Vitamin C (MS5.1)	OS/2&PC-DOS	£240
Power Screen (MS&TC)	PC-DOS	£105
Blaise View Mngr. (s'ce)	PC-DOS	£245
Lattice Curses	PC-DOS	£105
Entelekon Windows (s'ce)	PC-DOS	£105
Multi-windows (MS, L)	PC-DOS	£190
Panel Plus II (source)	PC-DOS	£270
Vitamin C (source)	PC-DOS	£165
Vermont Views (MSC, TC)	PC-DOS	£345
Greenleaf Datawindow	PC-DOS	£180
Greenleaf Makeform	PC-DOS	£ 60
C-Scape (+s'ce&Look&Feel)	PC-DOS	£330
Zortech Windows (ZC++&ZC3)	PC-DOS	£ 39
Curses/PC w. s'ce	PC-DOS	£265

ADA COMPILERS

The new AdaStudent is an affordable way to learn this important language.

AdaStudent	PC-DOS	£ 75
Janus/Ada compiler	PC-DOS	£ 355
IntegrAda	PC-DOS	£ 645

FORTRAN COMPILERS

The latest Prospero Fortran now supports OS/2. A new version of Lahey's compiler with improved optimisation now here.

Lahey F77L v4.0	MS-DOS	£450
Lahey F77L-EM/32	PC-DOS	£670
Lahey Personal Fort-77	PC-DOS	£ 75
FTN77/386 Fortran	PC-DOS	£695
RM/FORTAN 77 v2.43	MS-DOS	£450
MS-FORTAN 77 v5.0 OS/2&MS-DOS	£250	
Prospero Fort 2.10 OS/2&MS-DOS	£240	
Pro Fortran for GEM	MS-DOS	£105
Prospero PC Fortran	PC-DOS	£105
Utah Fortran	MS-DOS	£ 35
FS-Fort. (CGA & Herc)	MS-DOS	£ 34
SVS Fortran 386	PC-DOS	£575
Watfor 77	PC-DOS	£290
Pro Fortran v2.1	CP/M-86	£240
Pro Fortran v1.25	CP/M-80	£240
Nevada Fortran v3.3	CP/M-80	£ 35
Prospero Fortran (GEM)	ATARI ST	£105

DEBUGGING TOOLS

Logitech Multiscope	OS/2	£220
MagicCV	386&PC-DOS	£210
Periscope I/512K	PC-DOS	£580
Periscope II	PC-DOS	£115
Periscope II-X	PC-DOS	£ 95
Quaid Analyser	PC-DOS	£125
Soft-Ice	386&PC-DOS	£390
Soft-Ice + Magic CV	386&PC-DOS	£535

GENERAL FUNCTIONS

C+0 Class Lib (MS, TC)	PC-DOS	£155
C+0 Cl's Lib (MS) WINDOWS&PC-DOS	£155	
C/Utilities T'chest s'ce	PC-DOS	£ 39
CQL s'ce	PC-DOS	£270
C-Worthy Int.Lib (L, MS)	MS-DOS	£175
Code Runner	PC-DOS	£105
Faircom Toolbox Prof	PC-DOS	£665
Greenleaf Functions s'cePC-DOS	£105	
G'leaf Super Functions	PC-DOS	£135
G'leaf Business Mathlib	PC-DOS	£180
The Heap Expander (MS&TC)	PC-DOS	£ 65
Smorgasbord C-food	PC-DOS	£ 65
C Tools Plus/6.0 (MS)	PC-DOS	£105
Power Search	PC-DOS	£105
Turbo C Tools v2	PC-DOS	£105
Essential C Utils s'ce	PC-DOS	£145
Entelekon Funct (source)	PC-DOS	£105
Pro-C & Workbench	PC-DOS	£300
Resident-C (MS, L) (s'ce)	PC-DOS	£145
Silverware C EMM Lib	PC-DOS	£115
WKS Library	PC-DOS	£169
Zortech Supertext	PC-DOS	£ 39

COMMS LIBRARIES

C Asynch Manager V3s'ce	PC-DOS	£135
Essential Comms (L, MS, T)	PC-DOS	£195
Greenleaf Comms (source)	PC-DOS	£135
Multi Comms (L, MS)	PC-DOS	£190
Silvercomm C Asynch	PC-DOS	£145
Zortech Comms (QC&TC)	PC-DOS	£ 65
Zortech Comms (ZC++&ZC3)	PC-DOS	£ 65

SCIENTIFIC LIBRARIES

C Lang Sci Lib v2	ANY C	£175
Mathpak 87 (L, MS)	MS-DOS	£ 85

PROGRAMMERS UTILITIES

PC-Lint	OS/2, MS-DOS	£ 85
C-Documentor	PC-DOS	£195
C-Scan	PC-DOS	£195
Clear+ for C	PC-DOS	£165
Lattice Comp Companion	MS-DOS	£ 65

REAL TIME & MULTI-TASKING TOOLS

Concurrent C (PC/MPX) s'ceMS-DOS	£ 60	
Multi C (L, MS, CI, TC)	PC-DOS	£190
Op.Sys. T'box (MS) s'ce	PC-DOS	£ 65
Timeslicer v3.01 (L)	PC-DOS	£175
Timeslicer v5 (MS)	PC-DOS	£175
Over-C (L, MS)	PC-DOS	£225

PRICES & DELIVERY

Prices do not include VAT or other local taxes but do include delivery in UK and Europe. Please check prices at time of order, ads are prepared some weeks before publication.

This page lists some of our products. Call us for a complete pricelist.

Order by phone with your credit card.

GREY MATTER

2 Prigg Meadow, Ashburton, Devon TQ13 7DF

TEL. (0364) 53499

GREY MATTER

2 Prigg Meadow, Ashburton, Devon TQ13 7DF

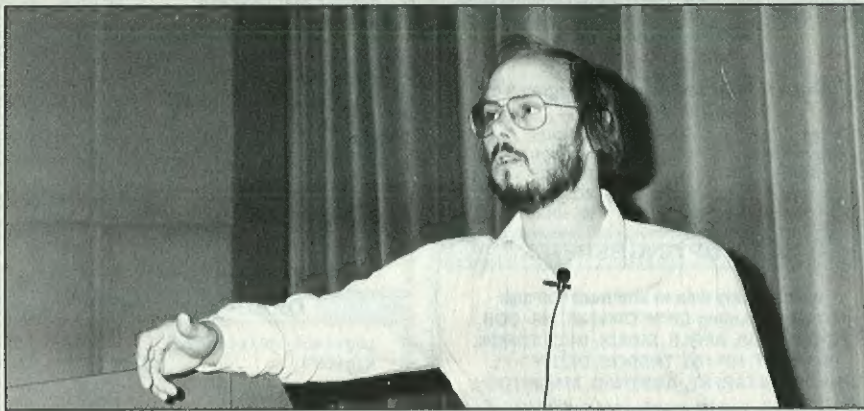
TEL. (0364) 53499

GREY MATTER

2 Prigg Meadow, Ashburton, Devon TQ13 7DF

TEL. (0364) 53499

C++ Seminar a success



On the 12th June 1990, in London, AT&T attracted 154 members of the European software industry to 'C++: A Technical Seminar for the 90s', a one-day event sponsored by .EXE Magazine. The keynote speaker was Bjarne Stroustrup, the designer and original implementor of the C++ language.

Stroustrup spoke about his aims for C++, summarised the history of the language, listed the key characteristics of C++, described the new 2.1 release of the language, and summarised the relationship of C++ to C and the recently begun standardisation process, and introduced two extensions to C++ that are currently being worked on: parameterised types, and exception handling. He made it clear that C++ is a systems programming language, enumerating its many benefits, but admitting its inadequacies in areas such as knowledge engineering and artificial intelligence research.

Other speakers at the seminar included Dag Brück of the University of Lund, Sweden, talking about C++ as a research tool; Bill Chown of Mentor Graphics, which has, over the last four years, developed more than two million lines of C++ code; Steve Scrase of Solbourne Computer, talking about Solbourne's OI object library; Paul Ivin of Sun talking about their C++ debugger; and Steven Carter of the C++ standardisation committee. John Carolan of Glockenspiel Ltd gave his views on the pitfalls and potentials of the developing C++ market. The seminar concluded with a panel session, in which members of the audience put their questions to Stroustrup and the other speakers.

At the seminar, Stroustrup announced his new book on C++: *The Annotated C++ Reference Manual*, by Margaret Ellis and Bjarne Stroustrup, published by Addison-Wesley, ISBN 0-201-51459-1. It should be available by the time you read this issue.

This Summer will be a good one for OOP conferences. There also is Tools '90 in Paris, from the 26th through the 29th of June, and Scoop Europe in London from the 16th through the 18th of July. Scoop Europe is joint sponsored by .EXE Magazine: see you there.

Paul G Smith

Tartan Ada

The US-based Ada compiler developer is offering a limited number of free Ada compilers to deserving colleges and universities of any nationality. The compilers run on VAXs (VMS and Ultrix) and Sun-3 workstations. Details from Tartan on 0101 412 856 3600.

Software Theft

A survey of senior British managers carried out by MORI on behalf of FAST (Federation Against Software Theft) found that 55% of the ones that used PCs admitted to copying software illegally. FAST estimates that over £300 million per annum is lost through illegal copying in the UK. FAST may be contacted on 071 240 6756.

UNIX Stars

We mentioned before that Ritchie, D was attending the UK UNIX systems User Group's technical conference (11th-13th July); it turns out that other guests include Thompson, K and Kernighan, B. Get your copy of the White Book autographed! Three day attendance costs £373.75 inc VAT to UKUUG members, membership costs £85.25. Contact 0763 73039 for further details.

Bye Bye floppy

The 5.25" floppy diskette is on its way out - it's official. The MIRC Report 'World Data Storage Markets' states that unit shipments of 3.5" sized drives are now outstripping those of the traditional 5.25" drives. The report goes on to predict the emergence of 2-2.5" drives, particularly in lap-tops, and the rise of optical and DAT technology. The report costs US \$1495, MIRC is on 0101 415 961 9000.

Legal Aid

City law firm Titmuss Sainer & Webb has produced a short guide to counterfeiting and piracy: the legal definition of these things and possible courses for redress if it happens to you. The guide is general - not software-oriented - but it is free on application. TSW's number is 071 583 5353.

Greek

If you want to run your Windows application in Greek, then you need a copy of DigiDuit. This will give you a choice of nearly 300 scaleable fonts (which you will have to buy as extras; you only get eight with the basic kit) which you can load into Windows and print on HP Laser compatible printers. Versions are also available to work with Ventura, Pagemaker, MS Word and WordPerfect V5. The supplier is David Pollard Associates (0865 240048).

PDOS

PDOS is a real time operating system from Derby-based Eyrisoft. Version 4.0 has just been launched, featuring a new file system, a POSIX-compatible file interface, network support and increased system modularity.

The file system is described as a 'forest': it consists of multiple device names (which may represent, for example, hard disk partitions or network interfaces) each of which is the 'root' of a conventional hierarchical directory system. File hooks have been provided to support 'foreign' file systems, such as MS-DOS and UNIX disks.

The software is available immediately. There will be free updates to those who purchased PDOS in the last 12 months, and to those who kept their software support up to date. Eyrisoft is on 0332 384978.

Novell Netware

Novell has produced V3.1 of Netware 386, its 32-bit real time operating system. Improvements over the previous release include open server APIs; a faster (claimed 25-30%) file system with support for third party drivers for removable media, such as CD-ROM and WORM; multiple name space support for different systems, so, for example, DOS users can access a 32-character Macintosh file with an 11-character DOS filename; disk mirroring increased to a maximum of eight mirrored drives; extended management services allowing detailed monitoring and controlling of network services and server resources; and EISA support.

The software should become available in July, from Novell's distribution channels. These can be determined by calling Novell UK Ltd on 0344 860400.

S O F T W A R E S E C U R I T Y

With £1 Billion Worth Of Protected Software...

SentinelPro™



- Runs under DOS, OS/2 and Xenix • Algorithm technique (Never a fixed response) • External parallel port installation • Minimal implementation effort • Higher level language interfaces included • 100 times faster than fixed-response devices (1 ms) • ASIC design for reliability

Sentinel-C™



- Protects multiple packages with one device • 126 bytes of non-volatile memory programmed before shipment of the software • Rainbow supplies a unique adapter for programming the unit • Higher level language interfaces included • Runs under DOS, OS/2 and Xenix • External parallel port installation

Eve™



- For the Macintosh SE and II • Complies with Apple Desktop Bus Interface requirements • Rainbow-assigned developer passwords to prevent tampering by other developers or sophisticated "hackers" • 7 locks per key, usable individually or in combination, on one or up to seven applications

SentinelShell™



- Runs under DOS on IBM PCs and compatibles • Protects without requiring access to the source code • Completely transparent to the end user • User-friendly software • Pocket-size key attaches quickly to any standard PC parallel port • ASIC design for reliability

DataSentry™



- Completely user-installable • Pocket-sized external device • Menu-driven, user-friendly interface • Single- or multi-user security system • Audit trail, log-on identifiers and automatic encryption/decryption of entire directories • Secures data transmitted by modems • Prevents recovery of data by utility programs

Rainbow Is The Safe Bet.

Software developers creating the latest applications for the IBM PC/XT/AT, PS/2 and compatible systems can now turn to the Software Sentinel range of hardware keys for the *first-class* in *world-class* software protection.

There's the best-selling SentinelPro, known worldwide for its virtually unbreakable security, its ASIC technology and its invisible operation.

A close relation, the Sentinel-C for custom configurations, enables multiple package protection with a single device.

For the Apple market, security-minded Mac software developers can now secure their return-on-investment, too. Eve plugs into the Mac ADB connector and is completely transparent to the user—providing up to seven programmable security locks per key.

Rainbow's latest protection strategy is the SentinelShell—that lets users place a "shell" around existing, off-the-shelf programs. Access can be limited to those issued a key, so that libraries, educational establishments and corporations can very simply guard their investments.

Available soon from Rainbow is the DataSentry, a user-installed key that provides low cost security for sensitive data in both database applications and corporate/banking environments.

No matter where you sell your software worldwide, stay in control of your distribution and revenue by choosing the internationally accepted standard in protection... Rainbow Technologies. Be sure. Protect your pot of gold at the end of the rainbow.

CALL TODAY OR ORDER NOW.

- ☐ Please send me a SentinelPro Evaluation Kit.
I enclose £50 + VAT Payable to Rainbow Technologies Ltd.
- ☐ Please debit my credit card. Access ☐ Visa ☐ Amex ☐
Expiry Date (PLEASE TICK)
- ☐ Please send me more information.

Name

Position

Address

Telephone

Signature EXE 6/90

RAINBOW TECHNOLOGIES LTD., Shirley Lodge, 470 London Road,
Slough, Berkshire SL3 8QY.



Shirley Lodge, 470 London Road
Slough, Berkshire SL3 8QY
Tel: 0753-41512 Fax: 0753-43610

Shrouded in Mystery

It would be great to develop a general-purpose UNIX application, because there are so many potential customers. But since there is no binary standard for UNIX, one would have to supply C source code. If you supply C source code, you are relying rather heavily on your customers' honesty.

Gimpel Software, which I have mentioned before for its *lint* product, has come up with an application called 'The C Shroud', a 'source code obfuscation tool for the C programming language'. You write your program in the normal way, just packed with helpful comments and meaningful identifiers (word of your brilliant code style has reached the .EXE offices). When ready to ship, you pass it through the C shroud. Here is a piece of code pre-shrouding (left) and after the treatment (right).

<pre>#include "stdio.h" #include "common.h" main() { int ch; while ((ch=getchar()) != EOF) output(ch); }</pre>	<pre>#include "stdio.h" void a286484(int); main() { int a72796; a117876:if ((a72796=getchar()) != EOF) goto a106688; goto a77946; a106688: a286484(a72796); goto a117876; a77946: ; }</pre>
--	---

The obfuscation includes comment removal, the conversion of high-level control structures into their `goto` equivalents, expansion of macros, identifier translation to randomised names and repetitive use of the same identifier sequence in non-overlapping name spaces (variables and structure tags, for example). All these features can be switched on and off as appropriate, so that the identifiers of library functions, such as `getchar()` in the example above, are left untouched.

The C Shroud is available for MS-DOS and OS/2 priced US \$198. Call the manufacturer direct for orders, and for prices for other machines. Gimpel's number is 0101 215 584 4261.

Black, Magic Box

Matrix Layout is an application generator for MS-DOS. You create a flow-chart, using a mouse-oriented, graphical interface; it spits out the application as BASIC, C or Turbo Pascal code. *Matrix Layout BlackBoxes* are function libraries which let you interface this system with dBASE files. These have just been introduced in the UK, and are being given away free with *Matrix Layout V2.0* (£169 + VAT + delivery). Call Matrix Software Technology (0752 796363) for more details.

Microsoft Project

Microsoft Project for Windows is a new graphical project management tool. It is completely GUI in design, and naturally you can cut and paste from the likes of Excel and Word for Windows. There are also links to dBASE and Lotus 1-2-3. The RRP is £495, upgrades from other versions of Project are offered at £120. The product should be available from dealers now.

OS/2 EE

IBM has announced the availability of OS/2 Extended Edition V1.2 (the extensions consist of the Communications Manager, the Database Manager and the LAN Requester). This is the version that doesn't use the 80386 properly - for example, you can only run one MS-DOS application, which drops dead as soon as you switch to an OS/2 task - and so doesn't look particularly attractive as a GUI compared to Windows 3. Especially as it costs £611.

EuroBIOS

Eurosoft Ltd, an independent European company, has announced the introduction of a 486/AT BIOS to complement its existing range of BIOS programs. Novel features of the BIOS include 32-bit data transfers between extended and conventional memory and a driver which performs EMS V4.0 emulation with extended memory. The company can supply the code in both ROM and binary file form, and it can be customised to your particular application. Details on 0202 297315.

Czech Mate

We quite often get puffs for programmer's tools in the form of examples of famous applications built with them. Normally we ignore this material, but this story is quite amusing: the Czechs used FoxBASE to collate the results of their first General Election, held in June. Bet you thought it was just for sorting the Customer Order file.

C++/Views

C++/Views is a class library specific for Microsoft Windows (although it is intended, like Glockenspiel's CommonView, to be generic; versions for OS/2 PM, the Mac and X are promised). It contains over 60 classes including basic 'foundations', such as containers, collections, sets, dictionaries etc, persistent objects (implemented through an Archiver), serial communications and, naturally, lots of user interface objects such as dialogues, controls and so on.

But there is more. C++/Views contains some C++ development tools: the best is a class browser. This can handle single and multiple inheritance, source editing, multiple rooted hierarchies, adding and deleting object classes, member visibility options, application management and (it says here) much more. There also are tools for building dialogue objects and the aforementioned archiver.

If all this works even half as well as the sales-literature suggests, then this should be quite an interesting package. To get it to go, you will need a Windows-compatible C++ V2.0 compiler (ie Glockenspiel or Zortech) and a Windows SDK. The product costs US \$495, and at time of writing is available only direct from its makers, CNS, in the US (0101 612 944 0170).

OOP Database

Ontos for OS/2 is an object-oriented database system. You can access it like a relational database, with embedded lines of SQL in your code. If you are working in C++, however, another avenue is open to you. Ontos supports persistent objects. By overloading the `->` operator, it is able to conceal much of the detail of the database operation.

The system works a bit like demand paging virtual memory. You write your data-manipulating code as though all the data that you require has been brought into memory. In fact, at each reference to a database object, Ontos's library code is called. If the data is already in memory, fine. Otherwise the database is searched for the relevant information. Naturally, it's not quite as simple as this; you do have to make various initialisation and transaction update calls as well, but the interface is much less cluttered than is usual for a DBMS.

Ontologic is also pushing a greater ability to handle complex data structures - for example, pointers and arrays may be included in the persistent objects - and a claimed performance improvement over conventional databases. More details from Ontologic on 0101 617 272 7110, or UK distributors Valbec Ltd (061 440 8231).

An object lesson in programming

C++ Meet Your Objective

Object-Oriented

Programming (OOP) is programming for the '90s. It's the next step after structured programming and is a much more productive way of writing applications. Borland has combined the power of OOP with the efficiency of C to produce new Turbo C++ Professional®.

Turbo C++ Professional is the first Turbo-charged native code C++ compiler that brings Object-Oriented Programming to your PC. Since Turbo C++ Professional also compiles ANSI C code, you can be productive with C now, and move to C++ at your own pace.

Environment ++

The best compiler deserves the best environment and our new Programmer's Platform® turbo charges your productivity. It features overlapping windows and mouse support, as well as a new multi-file editor, an integrated debugger, and a smart project manager. Its advanced open architecture lets you integrate the tools you need and are familiar with.

VROOMM adds room

VROOMM® (Virtual Run-time Object-Oriented Memory Manager) lets you break through the 640K barrier. Just select the application code you want to overlay, and VROOMM does the rest – swapping modules on demand. It's fast, easy, and automatic.

Another +

Turbo C++ Professional gives you all the tools you need to build fast, reliable C++ programs.

Turbo Debugger® 2.0 debugs your object-oriented programs. This powerful new version is the first and only debugger to support reverse execution, letting you step backwards through your code to find the bugs



you might have missed.

New Turbo Profiler®, the world's first interactive profiler, gives detailed analysis of your program's performance. With it, you can easily spot execution bottlenecks and see where improvements or redesign of your code will yield maximum performance gains. And Turbo Assembler® 2.0 lets you replace time-critical segments of your code using the world's fastest MASM® compatible assembler.

Buy objectively

The suggested retail price for Turbo C++ Professional, including Turbo Debugger, Turbo Assembler and Turbo Profiler is £249.95 plus VAT. Turbo C++ costs £149.95 plus VAT. For further information complete the coupon and return or call our PRODUCT INFOLINE on (0628) 771070 or simply talk to your dealer.

Turbo C++ Compiler

- C++ conforming to AT&T's 2.0 specification
- C++ class libraries
- Full ANSI C compiler
- VROOMM overlay manager
- Complete documentation and tutorials

Programmer's Platform

- Open architecture for integration of your own tools
- Overlapping windows with mouse support
- Multi-file, macro-based editor
- Smart project manager provides visual MAKE
- Integrated debugging and hypertext help

YES! Please send me a free Turbo C++ demo disk. ☐ 3 1/2" or ☐ 5 1/4"

NAME: _____

POSITION: _____

COMPANY: _____

ADDRESS: _____

POSTCODE: _____

TELEPHONE: _____

Please return this coupon to: Borland International (UK) Limited, FREEPOST RG1571, Twyford, Berks RG10 8BR

BORLAND

CIRCLE NO. 063

All Borland products are trademarks or registered trademarks of Borland International. Other brand and product names are trademarks or registered trademarks of their respective holders. Copyright © 1990 Borland International Inc.

Another New Product from the makers of Soft-ICE & MagicCV

ANNOUNCING BOUNDS-CHECKER

Gives you the protection of a protected operating system under MS-DOS.

AUTOMATIC PROTECTION

Flush out those Nasty pointer problems and other out-of-bounds memory accesses — AUTOMATICALLY.

Each time you make a change to a program, run BOUNDS-CHECKER while testing the new code. If you accidentally access out-of-bounds memory, BOUNDS-CHECKER will pop up displaying the offending SOURCE LINE. And your program runs at full speed.

***Increase Productivity
During Development,
Improve Reliability
During Q/A***

You can run BOUNDS-CHECKER while testing your program. There are no additional steps to your testing cycle, but you can feel secure when the program has passed through BOUNDS-CHECKER with no reported problems.

Many over-write problems and other out-of-bounds memory accesses do NOT show up during normal testing. An out-of-bounds memory location may be modified, but that particular location doesn't happen to be important at the time. Once the program is in the field and a certain network is loaded or a certain T&SR or device driver is loaded, that memory location suddenly becomes very important... AND THE SYSTEM CRASHES.

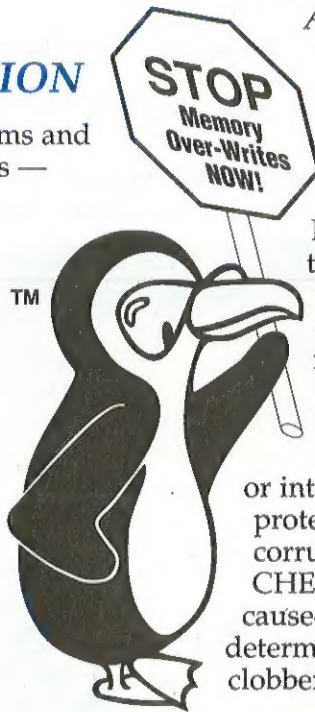
You can prevent these problems by making BOUNDS-CHECKER a standard part of your testing procedure.

Requires 80386 PC.
MS-DOS is a trademark of
Microsoft Corporation.

Finds out-of-bounds memory accesses — AUTOMATICALLY.

HOW IT WORKS

BOUNDS-CHECKER uses the 80386 virtual machine technology to provide real-time memory protection. In addition BOUNDS-CHECKER uses the symbolic information output by your compiler to differentiate CODE and DATA. When your program is running, BOUNDS-CHECKER protects the program's CODE and all memory outside your program. When an MS-DOS system call, BIOS call or interrupt occurs, BOUNDS-CHECKER protects the system software from corrupting your code. So, BOUNDS-CHECKER will not only detect problems caused by your program, it will also determine if a T&SR or other program is clobbering you.



Enormous Productivity Gain

Don't spend days or even weeks chasing down one of those subtle memory over-write problems. Use BOUNDS-CHECKER to pinpoint the problem in *seconds*.

Don't RISK a Recall

Don't take the unnecessary RISK that your program is unknowingly clobbering an out-of-bounds memory location. BOUNDS-CHECKER was developed to find these potential time bombs quickly, painlessly and AUTOMATICALLY!

\$249.⁰⁰

 **Nu-Mega**
TECHNOLOGIES

CALL TODAY 0101 603 888-2386 or FAX 0101 603 888-2465
P.O. BOX 7607 ■ NASHUA, NH ■ 03060-7607 ■ U.S.A.

CIRCLE NO. 064

Mercurial dBASE

Quicksilver/UNIX is a dBASE (III+ and IV) language-compatible compiler that has wandered into Fox Software's lair: it runs under PC versions of UNIX, such as SCO UNIX. The compiler produces native executable code, which can be executed without a run-time environment. According to the distributor, it is also dead quick. Data files are binary compatible with MS-DOS dBASE files; all you need to do is copy them from one system to the other. Extra features include a text windowing system, 512-field 4000 byte records, arrays with up to 255 dimensions (prize competition: a .EXE T shirt to anyone who can describe a sensible application for a > 10 dimension array; entries should be sent to The Editor) and better record locking - as one would hope when porting from MS-DOS to UNIX. A symbolic debugger reduces the wrench from an interpretive to a compiled environment. Limitations include the loss of interactive commands such as EDIT, BROWSE and MODIFY STRUCTURE.

Quicksilver/UNIX is distributed by Willaire Electronics PLC (0494 465 234). The package retails at £795.

Lattice Out

Lattice has released its 80286 C development system for MS-DOS and OS/2. What makes this compiler unusual is that it incorporates a 286 DOS extender. Code that is developed with the extender does not need a run-time licence, so in effect the thing may be used just like an ordinary DOS compiler. The package also includes a source level debugger, an editor, a linker and the biggest 'give-away' library that you will see with a DOS compiler - I know because I looked at it when reviewing the company's standard C compiler. There is yet another feature to this compiler which you don't see often - and not as big a selling point, I fancy, as the other things that I have mentioned. It's dongle-protected.

The US price for the package is \$495. Lattice is offering various upgrade bargains to owners of previous versions of its compilers. You may want to hang on for the 32-bit 80386 Lattice C compiler, which is due out soon. Lattice's UK distributor is Roundhill Systems: 0672 84535.

Check it with QCheck

There is no way to test any non-trivial application completely, but a good start is to make sure that every line of code has been run at least once. QCheck is an MS-DOS utility that helps you do exactly that. You compile and link your program to produce a Microsoft-compatible .MAP file with source code line numbers in it. You run the program under QCheck (which operates like a debugger), and can use a hot-key system to display, at any time, the source with *unexecuted* lines highlighted. If it's an interactive program, you can adjust the test input until you have forced an occurrence of every case.

When you have finished, QCheck can produce a listing of the source with the untested lines marked - just the thing to wave under the nose of the QA man. The drawback, of course, is that QCheck lives

in the same memory as your application, occupying 39 KB + 17 bytes/line of test program, and so restricts the size of program that can be tested.

QCheck is produced by Seltek Software, and costs £155 + VAT. Seltek's number is 088 341697.

HP's C++

Hewlett Packard has produced a new development environment for C++. HP C++/Developer can be used in conjunction HP's SoftBench environment, or as a stand-alone X-Window application. You need the graphics support because C++/Developer includes a graphical class browser and class construction facility. There is also an automatic MAKE file generator, and a cross-referencer. Details of prices and platforms from HP on 0344 369369 (and quote reference 4363).

Hyphen

Here's a C library the like of which you will not have encountered. It's very easy to describe the action of Hyphenologist: you pass it a string containing a word, it returns an array containing pointers to the places in the word where it can be hyphenated. That's all. It's rather more difficult to do, though.

For one thing, it depends on the language that the word was written in. Hyphenologist supports *forty one* languages (you try naming forty one languages) including English (phonetic and traditional), French, Italian, Latin, Lithuanian, Greek and Russian. The package is intended for writers of word processing, typesetting and DTP packages; it is the nichiest example of niche marketing that I encountered in the software industry.

Good luck to Computer Hyphenation Ltd (0274 733317), whose product is supplied in K&R C source form and costs from £2500 upwards - it depends on how many languages you want.

Disc-Lock

Disc-Lock is an MS-DOS utility which password protects a PC's hard disk. When locked, it prevents the user from getting out of an application and into MS-DOS, however hard he may press Ctrl-Break or tries to shell out. Its intended for corporate and public service use, where PCs need to be protected from unauthorised use. Disc-Lock costs £78 from Visionsoft (0274 610503).

Zephyr

Zephyr is 'a fully-relational, FoxPro-compatible DBMS for non-programmers'. According to the distributor, Advantage (0242 224340), it blends the best features of dBASE, FoxPro, Oracle, SuperBase IV and Alpha Four. Cor! You don't have to take this on trust, however, as there is a demo version, limited to 120 records, available for £6.95. The full package costs £59.95.

Paradox Brief

TCC (0978 290167) has linked the Brief Programmer's Editor into the Paradox Integrated Environment. You can edit PAL scripts from within Paradox, moving directly to the line being debugged. It will also look after your indentation for you. TCC will charge you £159 for a copy of Brief thus modified; £40 extra for a PAL template editor as well.

It did not die

Owners of Turbo Prolog V2.0 may like to know that, despite Borland's abandonment of the product, development continues. The original devisers of the product, a Danish software house called the Prolog Development Centre, has now produced version 3.2. You can upgrade to this for about £55; and there are also toolkits and a news-letter to see. Call PDC on 010 45 36 72 10 22. Don't worry - their English is better than mine.

DPMI

The DOS Protected Mode Interface is a new commercial standard which is being set up between such key companies as Borland, Eclipse, Microsoft, Phar Lap and Quarterdeck. It will allow multi-tasking DOS applications to share extended memory, and is the follow-up to the VCPI standard for compatibility between EMS emulators and Extended DOS applications.

Trojan Zip

Bulletin board users should look out for a program claiming to be V1.2 of the popular archiving program PKZIP. This is a Trojan, and should not be used. There is no legitimate V1.2 of PKZIP - and never will be.

Letters

We welcome short letters on any subject that is relevant to software development. Please write to The Editor, .EXE Magazine, 10 Barley Mow Passage, Chiswick, London W4 4PH. Unless your letter is marked 'Not for Publication', it will be considered for inclusion on this page.

Bugged!

Sir,

In his article on the 80386 protection mode in the May '90 issue of .EXE, David Bailey raised an interesting point which I have found useful. However, a minor correction is needed to the program in Figure 1, in order to avoid a bug that can be potentially dangerous. The bug is caused by the program not attempting to save the second interrupt controller mask before calling the BIOS service INT 15h. The attempt to save the first interrupt controller mask (as the program does) is not sufficient to preserve the state of the interrupts. The following code should be included before and after calling the BIOS service to remedy the problem:

```
;Before calling the BIOS
;read the mask
IN AL, 0A1H
;save it
MOV byte ptr int.mask2, AL
; After calling the BIOS
MOV AL, byte ptr int.mask2
OUT 0A1H, AL
```

*Dr K S Sabri
Douglas Electronics
Gwent*

Unprintable BASIC

Sir,

I have used Microsoft BASIC V7.0 for about two months. I am, on the whole, delighted with the improvements over V6.0b. However, I have also discovered several problems, all of which I have referred to Microsoft without having to elicit a single useful answer.

In one case, a large, multi-module application, which works perfectly under V6.0b, crashes at random intervals into DOS with the message 'unprintable error'. This despite the inclusion of a catch-all ON ERROR routine in my source.

Another problem concerns the use of expanded memory within the QBX envi-

ronment, about which I have little to say beyond that it does not appear to work as claimed.

In addition, the LINK switch /exepack produces a .EXE file which locks my computer.

Letters, faxes and telephone calls to Microsoft produce a fixed standard reaction: someone looks up their database and quotes the contents thereof, irrespective of whether or not they are relevant to the specific question. No answer received over the last two months has had any useful content.

Further on the subject of Microsoft BASIC, I am still hoping the User Group BASIC conference will, one day, be reinstated. I booked for the one last year that was cancelled without explanation. I have been programming in BASIC for 10 years and have yet to discover a forum for the exchange of ideas in that language. Any suggestions?

*Michael Gilbert FRICS
KEL Computing
Bourne End*

Mr Gilbert included further details of his problems, which are too lengthy to be printed here. Interested parties should apply to the editorial office for a copy.

dBASE Guide

Sir,

Do you know of an on-line guide for dBASE III+, similar to Norton Guides? If so, please could you send it or let me know how much and where to get it?

*Ron Nixon
Sigma Software
Southampton*

We rang Norton UK who said that Simon & Schuster (0442 231900) were dealers for an on-line help package for dBASE III. When we contacted Simon & Schuster, however, their representative confirmed the existence of the package in the USA, but said that the

company had not yet reached agreement with Norton US to distribute it over here.

For the time being, however, you may like to know that Nantucket distribute a product called SOS Help which is intended to work with the similar Clipper.

Chip Problem

Sir,

There appears to be a bug in some, if not all, 80386 and 80386SX processors which I've encountered. It is manifested during segment manipulation, and does not appear to be documented in any Intel manual that I've seen.

Briefly, if a segment register is MOVd to a 16-bit general purpose register, the top 16-bits of the corresponding 32-bit register are corrupted. So, the instruction MOV AX, SS will corrupt the top word of EAX.

I haven't had the chance to check this with an 80486 processor.

*Rupert Goodwins
Alfa Systems
London*

Disgruntled Turbo

Sir,

I have been using Turbo Pascal for nearly a decade. Over the years, I have built up a considerable library of code. Recently we purchased a SQL C API. In my naivety, I thought I'd be able to link Turbo C modules and Turbo Pascal units. I can't, unless I use Turbo Pascal V4.0, for which Borland had a .TPU to .OBJ converter; and even then I can't change the heap manager because I can't get the TP4 runtime library source.

Borland's position seems to be: tough. I think this is madness. I produce what Philippe Khan is pleased to call 'mission critical' applications, and I want to use my libraries. I can't, unless I rewrite them all in C. It's simply shoddy; and the release of C++ makes convergence even harder.

Jason Fordham

EXE

5 out of 5 hackers prefer other software protection methods to Hardlock E-Y-E®



What hackers dislike...

Hardlock E-Y-E was designed using cryptographic principles. It took the experience and know-how of Germany's No. 1 in software protection and the leading edge technology of a US semiconductor company to create the ultimate software protection tool. Hardlock E-Y-E is based on a custom chip featuring secure algorithmic response rather than simple bit swapping or counting schemes.

What software developers like...

Hardlock E-Y-E combines all the features software developers require in a single product: algorithmic response to provide security and an optional non-volatile memory to allow custom configurations. FAST Electronic has made implementation of Hardlock E-Y-E in your software easy. Use HL-Crypt to protect .EXE or .COM files, or incorporate high level language interface routines in your software. The algorithm parameters and the contents of the memory can be programmed in seconds using our Crypto-Programmer card. This unique card guarantees that no one else can burn your original codes. Simply plug the card into any PC slot and start up your own Hardlock E-Y-E workshop.

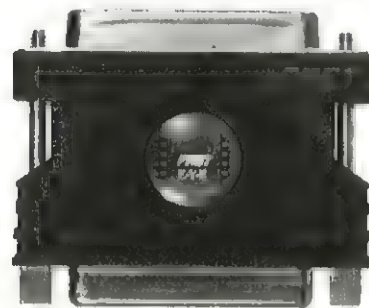
What your customers will like...

Hardlock E-Y-E allows unlimited backup copies. The device is shipped with the software for the user simply to plug into the parallel interface and forget.

Daisy chainability, outstanding reliability (no battery is needed), and the most compact High-Tech design ensure that your customer will accept Hardlock E-Y-E.

What your accountant will like...

Hardlock E-Y-E needs no factory coding. This ensures optimum delivery schedules and stock flexibility. Revenues will go up as software piracy and multiple usage are prevented. Despite its wealth of features, Hardlock E-Y-E's prices remain competitive.



...As more and more software developers, customers and accountants appreciate the Hardlock E-Y-E device, hackers like it less and less.

Hardlock E-Y-E
programmable, algorithmic response
and memory option - all in one.



**Order your demo unit today. Contact Magnifeye,
235-239 Walmer Road, Walmer Studio #6, W11 4EY, Telephone 071 221 8024, Fax 071 792 3449.**

Inside Windows 3

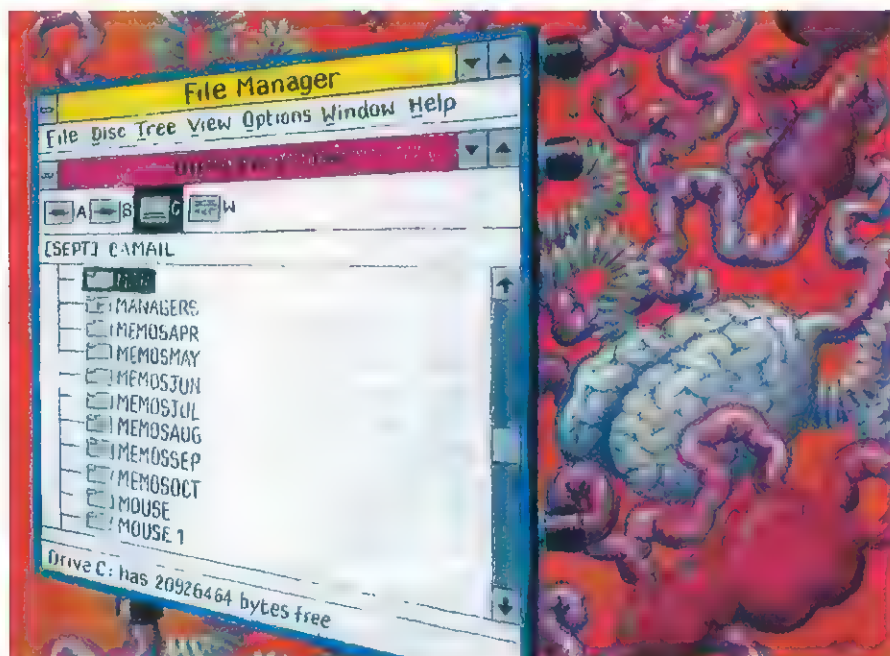
Handsome, bas-relief Microsoft's Windows 3 looks destined to become the wow of the Lotus 1-2-3 eating set. Peter Morris gives a technical introduction to its tricks and traps.

What's new in Windows version 3.0? Well, the biggest differences are in the types of memory, the use of both real and protected processor modes and a revised memory management system.

Whether your PC's RAM is conventional memory (ie directly addressable by the 8088/8086 processors), expanded (paged in memory) or extended (memory above the 1MB line, addressable by 80286-80486 processors running in protected mode), Windows doesn't mind. Plain vanilla DOS applications also may access all these types of memory from within Windows - essentially Windows utilises your machine's memory to the full, whatever it may be. There is enough subject matter in Windows' handling of these memory types, DPMI etc to fill a whole article, but, for now, lets move on to modes and memory management.

Windows 3 has three distinct modes of operation: real, standard and enhanced. Standard and enhanced modes work by running your 286/386 processor in its protected mode, and it's in this mode that up to 16 MB of extended memory becomes available. If you have a 386/486 machine, Windows' enhanced mode can also use disk space as virtual memory. At the other end of the scale, real mode, the processor (irrespective of what it is) is run in - you've guessed it - the real mode, allowing access to just 1 MB of address space.

DOS applications running under Windows may take advantage of all the different types of memory available in your PC (although some are only available in certain modes). If you're using a 386/486, each DOS session/application can be run in a separate virtual 8086 machine. These applications may be as badly behaved as they like; for example, they may attempt to write directly to video memory. Under enhanced mode, text mode DOS applications may



even be run in a window. As far as the application is concerned, it is running on an 8086 with 640 KB of memory (although extended, expanded and virtual may also be used, if the application knows what to do). Attempts to write directly to video memory are trapped. Windows runs a 'text mode emulator' in the window it has pro-

vided for the application (Windows itself, of course, runs in graphics mode), and it's here (and only here) that the application's output may appear. The system is very robust: you can install TSRs, and even cut and paste from the DOS window to the clipboard. What's more, these virtual machines may be pre-emptively scheduled - ie the PC

```
#define ONE_K 1024

GLOBALHANDLE hMem;          // memory handle LPSTR lpDemo;
// far pointer

hMem=GlobalAlloc(GHND,ONE_K); // get 1K

GlobalReAlloc(hMem,ONE_K,GMEM_ZEROINIT); // add 1K

lpDemo=GlobalLock(hMem);

... // access memory data here

GlobalUnlock(hMem); // implicitly invalidate pointer

GlobalFree(hMem); // implicitly invalidate handle
```

Figure 1 - Common misuse of GlobalRealloc()

can run several DOS applications (and keep them running using time-slicing techniques).

Programming

Time to have a look at some programming issues; in particular, how to write programs for protect mode Windows applications, and how to go about converting a Windows 2 application to work under Windows 3. I should start by saying that the Windows 3 API is a superset of Windows 2, so there are no functions which 'no longer work'. That is not to say, however, that Windows 2 applications are automatically upwards compatible.

Microsoft says that applications wanting to run under Windows 3 (standard and enhanced modes) should be 'protect mode aware'. At QA Training, we would say that such applications were simply API aware and conventional! For example, one of the changes that trips people up is to do with the routine `GlobalReAlloc()`, which reallocates an area of memory in the global heap. Here is what the manual says about `GlobalReAlloc()`'s return value:

'...the return value identifies the reallocated global memory if the function is successful. The return value is NULL if the block cannot be reallocated.'

Judging from this, it would seem essential to record the return value for subsequent use. However, under earlier versions of Windows, it was possible to ignore the return, as `GlobalReAlloc()` always passed back the value supplied to it in the argument `hMem`. Take a peek at Figure 1 to see how this was done.

This approach worked fine under Windows 2.x. The handle `hMem` never changed value during the call to `GlobalReAlloc()`, so the reassignment wasn't needed. Conscientious programmers may have used the return to check for an out of memory failure, but it certainly was not needed to access the reallocated memory block.

Many programmers discovered this 'implementation specific' feature (Microsoft's code examples use it all the time) and built reliance upon it into their code. This code will now need modifying, as the handle's

value is (sometimes) not maintained. (It all depends on whether the data segment crosses a 64 KB boundary during reallocation. In protect mode, this is catered for by a technique called 'selector tiling'). The correct technique is illustrated in Figure 2.

Another facet of 'Protect Mode Aware' programs is that they must not do segment arithmetic. In real mode, where far pointers comprise of a segment and offset, it is possible to bump a pointer forward 16 bytes by incrementing its 16-bit segment component. Under protect mode, far pointers aren't what they seem; any attempt to do something like this will result, at best, in a protection violation. This is because the segment part of a far pointer is not an address value, but a pointer to a descriptor table entry. It is this entry which holds the paragraph address of the object we are trying to point to. If you're familiar with the Windows 'thunk' mechanism, it's like having a reload thunk for data. Thus, an attempt to increment the segment part of a far pointer increments the pointer into the descriptor table. You can do this, but it is certainly not what you meant, and it is desirable to catch this situation. Protect mode is made for this sort of stuff.

Here is a further side effect of protect mode. Applications that need multiple data segments (none of them do, in point of fact) no longer have to lock them down. The whole notion of 'segment locking' is nonsense in protect mode - you just tell if a segment has moved, locked or not. This means that compact, large or even huge memory model applications may now be written without fear of

Windows 3 - the main changes

There is much better use of hardware - the CPU's protected mode, the various types of extra memory.

- There is a new control, called 'COMBO', which combines edit and list control.
- The existing controls have new functionality.
- There is a sophisticated Help manager, and a standardised Help file format.
- There are much improved colour-handling facilities, with a Palette manager
- There is an improved API, with more functions and greater capability.
- The new '3D' front end - the buttons now appear to move in and out when you click on them.

There is a new set of tools to go with Windows 3.
There are also other, minor changes.

system performance degradation - provided that they are used in protect mode only. By the way, Microsoft says that medium memory model applications should still be the norm, as your users may be using Windows 3.x real mode. Read on...

If Windows needs to move a data segment (we'll say of a multiple data segment application) it can now relocate the segment without worrying about the effect on your application. For example. You have a pointer to a piece of local data (by far reference) and Windows needs to move the data. All it has to do is change the entry in the descriptor table (referenced by your pointer's segment word) so that it points to the base address of your data segment. Your pointer has the same value as before, but the object to which it points has moved, relative to the real memory map. It's one of those 'frame of reference' conundrums.

One other thing I should point out here. You'll notice in Figure 1 that I've included comments such as 'implicit invalidation of pointer'. You should always *explicitly* invalidate pointers and handles to now defunct or extinct objects when you're through with them. There should also be routines to 'validate' these objects prior to their use. If you use this technique, it should be impossible for you to use invalid references or objects (either direct or indirect references).

To implement this idea, most serious Windows programmers devise their own 'home-made' versions of the allocation/locking, freeing and validation routines - mine are shown in Figure 3. The routines could be placed in their own DLL. They perform the function of the 'real' API routines (eg unlock some memory), but they also explicitly invalidate any defunct implicit pointer or handle. Before using a pointer or handle, you call a 'validate' to check that it's ok.

```
hMem=GlobalAlloc(GHND,ONE_K); // get 1K
hMem=GlobalReAlloc(hMem,ONE_K,GMEM_ZEROINIT); // add 1K
lpDemo=GlobalLock(hMem);
```

Figure 2 - Correct use of `GlobalReAlloc()`

FINALLY. A debugging tool tough enough to handle the DOS Nasties.

New Version 2.0



Nasty over-write? No sweat!

Soft-ICE memory range break points help you track down memory over-write problems whether you are doing the over-writing or another program is over-writing you.

Hung program? No problem!

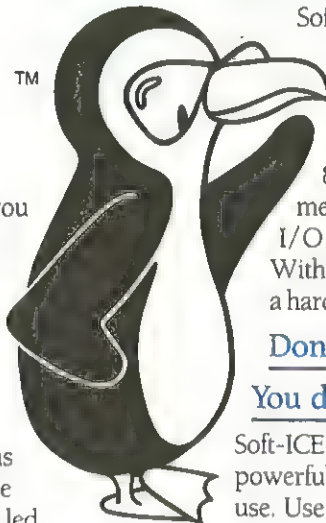
When the system hangs, you now have hope. With Soft-ICE you can break out of hung programs no matter how bad the system has been trashed. And with Soft-ICE's back trace ranges you can re-play the instructions that led up to the crash.

Program too large? Not with Soft-ICE!

Soft-ICE runs entirely in extended memory. This means you can debug even the largest DOS programs. And since your program runs at the same address whether Soft-ICE is loaded or not you can find those subtle bugs that change when the starting address of your code changes.

System debugging? Soft-ICE is a natural!

Soft-ICE is ideal for full source level debugging of TSRs, interrupt service routines, self booting programs, DOS loadable device drivers, real-time kernels, non-DOS O/Ss and ROMs. Soft-ICE can even debug within DOS & BIOS.



How Soft-ICE Works

Soft-ICE uses the power of the 80386 to surround your program in a virtual machine.

This gives you complete control of the DOS environment, while Soft-ICE runs safely in protected mode. Soft-ICE uses the 80386 to provide real-time break points on memory locations, memory ranges, execution, I/O ports, hardware & software interrupts. With Soft-ICE you get all the speed and power of a hardware-assisted debugger at a software price.

Don't want to switch debuggers?

You don't have to!

Soft-ICE can run stand-alone or it can add its powerful break points to the debugger you already use. Use your favorite debugger until you require Soft-ICE. Simply pop up the Soft-ICE window to set powerful real-time break points. When a break point is reached, your debugger will be activated automatically.

MagicCV with Soft-ICE

Using Soft-ICE with CodeView gives you the features necessary for professional level systems debugging. MagicCV and Soft-ICE can work in concert with CodeView to provide the most powerful debugging platform you will find anywhere.

"These may be the only two products I've seen in the last two or three years that exceeded my wildest expectations for power, compatibility and ease-of-use."

—Paul Mace
Paul Mace Software

Soft-ICE	\$386
MagicCV	\$199
MagicCV for Windows	\$199
Buy Soft-ICE & MagicCV(W)	—Save \$86.
Buy MagicCV and MagicCVW	—Save \$100.
Buy All 3	—Save \$186.

30 day money-back guarantee
Visa, MasterCard and
AmEx accepted



New Product / New Idea

Find overwrites and un-initialized pointers Automatically

All the protection of a protected O/S under DOS

Bounds Checker - \$249.

Nu-Mega
TECHNOLOGIES



CALL TODAY 0101 603 888-2386
or FAX 0101 603 888-2465

RUN CODEVIEW IN 8K

MagicCV



CodeView is a great integrated debugger, but it uses over 200K of conventional memory. MagicCV uses advanced features of the 80386 to load CodeView and symbols in extended memory. This allows MagicCV to run CodeView in less than 8K of conventional memory on your 80386 PC.

NEW—Version 2.0 includes EMS 4.0 driver.
Attention Windows Developers!
Version available for CVW.

P.O. BOX 7607 ■ NASHUA, NH ■ 03060-7607 ■ USA.

CIRCLE NO. 066

Figure 3 - Home-made memory allocation routines

```

BOOL ProtectLocalAlloc(WORD wFlags,WORD wBytes,HANDLE *hMem)
{
    return (*hMem=LocalAlloc(wFlags,wBytes)) != NULL;
}
BOOL ProtectLocalLock(void NEAR *pMem,HANDLE hMem)
{
    return ((*pMem=LocalLock(hMem)) != NULL);
}
BOOL ProtectLocalUnlock(void NEAR *pMem,HANDLE hMem)
{
    return ((*pMem=LocalUnlock(hMem)) == NULL);
}
BOOL ProtectLocalFree(HANDLE *hMem)
{
    return ((*hMem=LocalFree(hMem)) == NULL);
}
BOOL ValidateNearPointer(char NEAR *pMem)
{
    return pMem!=(char NEAR *)NULL;
}
BOOL ValidateHANDLE(HANDLE hMem)
{
    return hMem!=(HANDLE)NULL;
}

```

Handle Abuse

I have a great wrench for dyed-in-the-wool sloppy programmers. You really are going to have to stop casting memory handles to pointers. (You might also like to stop using C, but let's do one thing at a time.) Would you treat an integer like a real? Would you treat a cow like a dog (perhaps by feeding it beef offal)? Quite. So why treat handles as pointers? Come on, guys and gals! This is an object-oriented environment that we are using here. If you respect the implementa-

tion's privacy, your client code will never be broken.

Let's take a look at the sort of situation that leads us into temptation. Figure 4 shows the correct way to generate heap addresses, Figure 5 the wrong way. Needless to say, the code in Figure 5 works. (Where do you think it was first seen? Yes, you've guessed it, Microsoft code.) It's wrong because we are using knowledge about the implementation of a handle object to bypass the LocalLock() and LocalUnlock()

```

// LOCAL OBJECTS:

LOCALHANDLE hLocalMemory; // handle to local memory
NPWNDCLASS pLocalMemory; // pointer to local memory

// need something the size of a WNDCLASS from the heap.
hLocalMemory=LocalAlloc(LMEM_ZEROINIT,sizeof(WNDCLASS));

if(hLocalMemory)
{
    pLocalMemory=(NPWNDCLASS)LocalLock(hLocalMemory);
    // get address from handle by locking handle
}
else
{
    // deal with "didn't get local memory" error
}

// process memory here, for example
// pLocalMemory->lpfnWndProc=MainWindowWndProc;
// etc
//

// free up memory

LocalUnlock(hLocalMemory); // implicit invalidation of
// pointer pLocalMemory

LocalFree(hLocalMemory); // implicit invalidation of
// handle hLocalMemory

```

Figure 4 - Correct way to generate local heap addresses

calls. LocalLock() takes a handle to a memory object, locks it down, then returns the address of that object in the local heap (this 'lives' within the application's default data segment). In the 'implementation specific' code, the handle is de-referenced using our ill-gotten knowledge. However, the object must be prevented from moving - we must simulate the effect of LocalLock(). This is done by asking the local allocator to supply a 'fixed' object. In the Figure 5, there is a call

```

LocalAlloc(LPTR,...);
where LPTR = 'Local
Pointer' = LMEM_FIXED |
LMEM_ZEROINIT is set up in WIN-
DOWS.H.

```

WINDOWS.H also contains the definition of a GPTR constant, which suggests that the same dirty trick may be done on global objects. This is indeed the case, as you can in see Figure 6. You take the result of calling the global allocator and zero, turn them into a long, cast that to be a far pointer to WNDCLASS and assign the result to lpGlobalMemory.

You shouldn't do this, no matter what Microsoft does. Apart from the importance of treating objects as objects, and not as segment addresses, there is one real nasty thing about this allocation code. It's not checked for validity within the expression, or even after it's been initialised. Consider what will happen if the GlobalAlloc() request, which returns NULL on failure, goes wrong.

Try the Figure 6 trick while running under one of the Windows 3 protect modes, and you will find yourself looking down the wrong end of a GP violation (best case) or accessing the wrong global segment (worst case). Remember, the segment part of a far pointer is now a segment selector. It is theoretically possible to fabricate a far pointer from the selector and an offset, but if you try and use a memory object allocated by GlobalAlloc() without first calling the GlobalLock() function, a GP error is the best that you will get, and it's more than you deserve.

Marking

Applications written under Windows 2.x may be identified as being 'protect mode aware' using the Windows 3 SDK 'mark' utility. Normally, if you attempt to run an application created using an earlier SDK version, with Windows 3 in either standard or enhanced mode, you'll be warned that it isn't protect mode aware. (In fact, the actual message displayed says 'this application was written

Figure 5 - Wrong way to generate local heap addresses

```

NPWNDCLASS pLocalMemory; // pointer to local memory

// need something the size of a WNDCLASS from the heap.

pLocalMemory=(NPWNDCLASS)LocalAlloc(LPTR,sizeof(WNDCLASS));

if(pLocalMemory)
{
    // process memory here, for example
    // pLocalMemory->lpfnWndProc=MainWndProc;
    // etc
    //
}
// free up memory

LocalFree((LOCALHANDLE)pLocalMemory);

```

Figure 6 - Wrong way to generate global heap addresses

```

LPWNDCLASS lpGlobalMemory;

lpGlobalMemory=(LPWNDCLASS) (MAKELONG(0,
    (LONG)GlobalAlloc(GPTR,sizeof(WNDCLASS)));

//... use the pointer here

GlobalFree(LOWORD((DWORD)lpGlobalMemory));

```

for a previous version of Windows', but it really means that it isn't protect mode aware.) You are given the choice of either restarting Windows 3 in real mode (typing the command line 'win /r' does this) or pressing 'OK' to carry on regardless. A long study of human nature leads me to suspect that most people will take the latter course. If you have written a Windows 2.x application, and you know that it's protect mode aware, then you can mark it as such with 'mark', and the warning message will be suppressed. Nor will it appear if you develop your application with the Windows 3 SDK - but note that, as we have seen, use of the current SDK in itself does not prevent you from writing 'protect mode blind' applications.

Another conversion catch: Windows 3.0 uses a non-fixed pitch system font. If you have used spaces to line up the strings in your .RC files, then you will need to modify them. Figure 7 shows an example of this sort of problem. If the text strings are left as they stand, they would display something like this (whichever mode is used):

First item Ctrl+F
Second item Ctrl+S
Third item Ctrl+T

To get this right under Windows 3, you will need to align things using \t tabs:

```

MENUITEM "&Whatever\tCtrl+W",
    ID_WHATEVER

```

Windows 3.0 applications must be aware of the proportional font if a professional effect is to be obtained.

Portability

If you want to create an application which runs under both Windows 2 and Windows 3, you must develop it under Windows 2 and mark it. If you try to run a Windows 3 application under Windows 2, you'll be told that it requires a later version of Windows. It's the resource compiler that marks an application to be a certain version, so even applications that don't have resources should be processed by the resource compiler. You can use a .RC file containing

```
/* NO RESOURCES AT ALL */
```

compiled with the command 'RC EMPTYRES.RC', to create the

application EMPTYRES.EXE. This is sufficient to mark it as a Windows 3.0 application.

If you have it, the segmented executable linker supplied with the Microsoft C V5.1 compiler can be used instead of the LINK4

utility that comes with the SDK. The segmented executable linker is simply a later version of LINK4. If you do use it, however, you must include the following in your .DEF files:

```

NAME TEST
DESCRIPTION 'test application'

```

```
EXETYPE WINDOWS
```

It's the EXETYPE line that is important. Incidentally, you can add this line to all your old .DEF files if you like, as LINK4 ignore the EXETYPE directive.

Conclusion

There are other things that you need be know, but there is not enough space to tell you about them here. The key is remember to (keep banging the rocks together?) treat the API with respect and make your code conventional. Writing your applications in a different language than C would help - general-purpose programming my language my foot! (*That's quite enough whinging about C for this issue - Ed.*)

Perhaps an approach based on a UIL (User Interface Language) will come along soon, and make everything easier. It seems like common sense that these applications should be written in some sort of GUI language, and not by calling routines in a library. Meanwhile, my advice is to use anything that helps you master the ever-widening API and its associated programming problems.

[EXE]

Dr Peter Morris is the Senior Windows Lecturer at QA Training (0285 655888). He has been teaching Windows 3 for many months.

Microsoft Windows 3 costs £99, and is available from dealers now. The Windows 3 Software Development Kit will start shipping during July.

```

#include "style.h"

#define ID_ONE 1
#define ID_TWO 2
#define ID_THREE 3

TestMenu MENU
BEGIN
    MENUITEM "&First item" Ctrl+F, ID_ONE
    MENUITEM "&Second item" Ctrl+S, ID_TWO
    MENUITEM "&Third item" Ctrl+T, ID_THREE
END

```

Figure 7 - Unmodified resource file

CASE:PM or CASE:W

THE TOOLS THAT WRITE PM OR WINDOWS CODE FOR YOU!

Writing applications for Windows or OS/2 Presentation Manager graphical user interface just became a lot easier, because now there's CASE:PM (for Presentation Manager) and CASE:W (for Windows).



Both are application prototypers and code generators that write complete C skeleton programs for you based on a description of the 'user interface'.

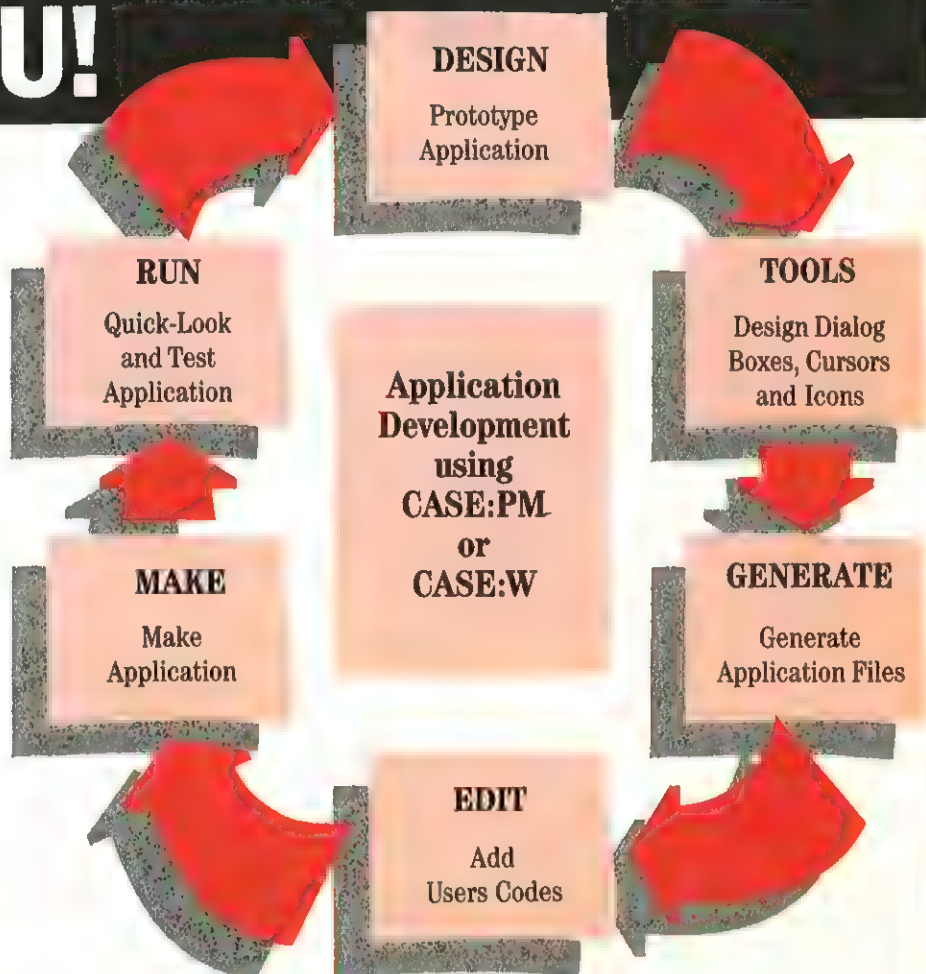
APPLICATION PROTOTYPING

Just by 'pointing and clicking' at various parts of a window you can now create a Windows or PM user interface – in minutes!

APPLICATION GENERATION

Once you have the 'look and feel' of the application, CASE:PM or CASE:W produces C source code, resource scripts, definition files and make files... *automatically*.

Run *MAKE* and the files are compiled and linked producing a standalone executable with no run-time environments or royalties to pay.



APPLICATION MAINTENANCE

Once you have the skeleton you add the application specific code. The generated source code is fully documented, with instruction as to where to put the application specific lines, leaving you free to concentrate on the application itself.

Change the user interface design and CASE:PM or CASE:W will generate the source files with the new user interface – but retaining the code you've added.

CASE:PM or CASE:W will reduce your learning curve and increase your productivity. For information and technical specifications call the Products Group on 0285 655888 FAX (0285 650537), or write to us.

QA

Cecily Hill Castle
Cirencester, Gloucester GL7 2EF
0285 655888

CIRCLE NO. 067

Red is the Colour of Turbo C++

Turbo C++ has ended its career as vapourware. Willie Watts has been taking a look at the latest compiler from the Borland stable.

Borland likes to see itself as the populariser of esoteric languages for the common programmer. Before Turbo Pascal, the Pascal language was an educational tool, running on college mainframes and the occasional Apple. The few PC translators that existed huffed and puffed to produce greasy, corpulent object code that sometimes ran almost as fast as the BASIC interpreter that came free with the machine. Post-TP, some of the reference books now describe Pascal as a 'real time language'; if that isn't Borland's fault, it's difficult to see who else to blame.

But wait. This has not been a story without a hitch. More recently, Borland introduced Turbo Prolog. This was a non-standard implementation of the language (doubters who pointed this out were brushed aside with cries of 'Turbo Pascal was non-standard, too') which was to be the first really pragmatic version of Prolog. Artificial Intelligence had been made accessible to the PC user. About six months ago, Borland announced that it was returning the rights of this product to its original owner. Owners of Turbo Prolog were left, if not entirely up the creek, then certainly a fair distance from the estuary.

Turbo C++ is different. It is, relatively speaking, much more important to the company's success: of its surviving lan-

guage product lines, Turbo C++ represents half. However, despite Borland's description of itself as 'the leader in object-oriented programming' this is its deepest foray into the territory (last year's intermediate release of Turbo Pascal carried a strong hint of anti-QuickPascal marketing, and rather less OOP technology). Will the company hold firm when the inevitable anti-OOP backlash begins? The spectre of Turbo Prolog has yet to be exorcised.

What you see

Turbo C++ comes in two flavours: 'straight' C++, which consists of the compiler, integrated environment and associated tools and the 'Professional' version, where you also get the latest version of Borland's deservedly famous debugger, an assembler, and a new profiler. These tools are also available separately and, owing to constraints of time and space, this is how I shall be reviewing them. This review, then, confines itself to what we might call 'Turbo C++ Amateur'.

Red, for those of you who have been wondering about the melodramatic headline at the top of this piece, is the dominant colour of Turbo C++'s box and manuals. The software comes on eight 360 KB disks (Microsoft now supplies 1.2 MB as standard, and invites those without AT drives to send off

for an alternative disk set) stored in compressed format. An easy-to-use and business-like INSTALL program moves the stuff onto your hard disk in short order; you get to nominate the target directories, which memory model libraries that you want and so on. An omission is that, unlike some rival's recent efforts, the program does not report exactly how much disk space your selection requires before it starts; it merely states that the maximum is about 6MB. The configuration that I chose - small and large library memory models only, plus everything else in the way of examples and tutorials - occupies 5 MB.

With everything gathered safely on board, time to load up the integrated development environment, which is still called TC.EXE, despite having been renamed the Programmer's Platform, if you see what I mean. The IDE (as I shall stubbornly continue to call it) has been entirely reworked. That influential Chinese-born designer See-Yu Ay has been let loose among the hallowed keystrokes and menus of Turbo C. Having been comparatively slow to acknowledge the onset of the mouse, Borland has now introduced it with vengeance.

Mouse-friendly means keyboard-hostile, at least to start with. You constantly find yourself hitting ENTER where you meant TAB, operations that used to take two keystrokes

	register int	auto short	auto long	int multiply	function	auto double	compile time (secs)	.EXE size
Lattice V6.0	0.31	0.35	0.98	1.23	1.61	27.1	26	27646
Microsoft V6.0	0.33	0.33	1.7	0.32	1.1	49	35	32310
TopSpeed V1.0	0.26	0.26	0.83	1.42	1.27	51.6	12	27590
Turbo C V2.0	0.35	0.38	1	1.2	1.1	110	6	27348
Turbo C++ V1.0	0.34	0.35	1.16	1.18	1.15	97.8	10	27552
Watcom V7.0	0.24	0.24	0.75	1.2	1.1	8.02	27	18119
Zortech V2.0	0.31	0.31	1.04	1.2	1.16	18.7	14	24160

Except for compile time and .EXE size, the results are shown in microseconds per individual operation, using a 20 MHz 386 machine.

Figure 1 - Plum Hall C Benchmarks, with compiles time and file size.

"The MKS Toolkit is an amazingly faithful replication of a System V UNIX™ environment."

- UNIX Review

Essential UNIX Tools for any PC user



"The entire MKS documentation package proved excellent in every respect"

Daniel McAuliffe, IEEE Computer, Jan. 1990

We can tell you that MKS Toolkit offers both experts and novices the purest form of UNIX utilities in a DOS or OS/2 environment. Fortunately, we don't have to. Software reviewers, universities and major corporations all over the world are discovering MKS Software and how it can help their programmers and software developers make the most of their creative talents.

With our proprietary code, MKS Toolkit offers you more than 150 UNIX System V, Release 4 compatible tools for DOS or OS/2. With MKS Toolkit, your computer or clone becomes a comfortable environment for shells, string matching, editing, file manipulation, and more.

In this one package you will receive a complete:

- MKS Vi full – screen editor
- MKS Korn Shell – user programmable interface

- MKS AWK – programming and prototyping language
- profiler – to analyse the execution time of code
- egrep/fgrep – fast pattern matching
- archivers, sort/merge, encryption, data compression & much more

Over 150 commands that help you get the most productivity out of your PC.

Easy Installation

With MKS's menu-driven installation you can be up and running in less than 10 minutes, no matter what your level of computer expertise.

"MKS software is absolutely the best in its class. Don't mix environments without it."

Grover Righter, Novell

Interconnectivity & Lan Licences

MKS Toolkit interacts well on most standard PC and PS/2 networks. Combined with Novell Netware™, the most popular LAN for PC's, MKS Toolkit creates a UNIX time sharing system in DOS or OS/2 organizations. UNIX workgroups can now hook up all of their PCs using PC-NFS™ and MKS Toolkit, enabling you to use a PC as a UNIX workstation and off-load your mini or main machine.

Spanning Novice to Expert

This versatile software reflects it's users' needs. UNIX experts will have all the familiar power they demand while novices will find our documentation filled with excellent tutorial material.

Organizations such as AT&T, Hewlett-Packard, ITT, and NCR – all heavily committed to the UNIX system –

use MKS Toolkit to create a standard operating environment. Universities from King's College to the University of Berlin use MKS Toolkit to enrich personal research computing environments and double the potential of their PC teaching labs.

"With this package, you can become familiar with the UNIX environment on your microcomputer, with DOS only a keystroke away."

Byte Magazine, May 1989

Cost-Effective Learning Tool

If your organization is committed to moving into the UNIX environment, then MKS Toolkit is the perfect learning path. DOS or OS/2 users retain the familiar world of their PC keyboard and programs yet can move to a desktop UNIX environment at a keystroke. Exposure to new commands and functionality now becomes an integral part of the novice's working day.

MKS Programming Platform

MKS Toolkit can be purchased as an individual product or you can buy MKS Programming Platform which contains MKS Toolkit and three other MKS products. MKS Platform helps smooth out the details of programming and software development by adding time-saving utilities such as: MKS RCS (Revision Control system), MKS Make (an automated program builder), MKS LEX & YACC (compiler learning and construction tools).

In all, you simply cannot find a more complete set of commands and utilities to get from DOS or OS/2 to UNIX. Once you have used MKS Toolkit, programming will never be the same again.

US List Price

MKS Toolkit (DOS & OS/2)	\$399.00
MKS Toolkit (DOS/2 only)	\$349.00
MKS Toolkit (DOS only)	\$249.00

Please contact MKS dealers for UK domestic prices.

The Software Construction Co. Ltd. (0763) 244114; fax (0763) 244025
Grey Matter (0364) 53499
Systems Science (071) 833-1022; fax (071) 837-6411



35 King Street North, Waterloo, Ontario, Canada N2J 2W9

Put The Power Of UNIX In Your PC.

MKS, MKS Make, MKS Toolkit Korn Shell, and MS RCS are trademarks of Morrice Kern Systems Inc. UNIX is a trademark of AT&T
PC-NFS is a trademark of Sun Microsystems

CIRCLE NO. 068

(loading in the file-before-last, for example) now seem to take loads. There are some neat new features - a 'history' function lets you step back through your previous selections of search/replace and file selection mask strings - but I am sure that the net change is an increase in clumsiness of use. I am uncomfortably aware of the high subjective element in this judgement, and apologise for it. My guess is that current users of Microsoft Quick... languages will like it better than current users of the Turbos.

Anyway, at least the default editing keys are still Wordstar (function key assignments have changed slightly), you may tamper with the setup if you wish. The editor supports a simple macro language - but you have to compile text files with a separate macro compiler; there is no macro recording facility. You can use this language to 'wire in' other programs into the environment. (If you get really keen on the IDE, you can even write a filter program to let it track error messages output by another compiler). There is now a clipboard mechanism for manipulating blocks of text - it is half-way between the Wordstar approach (where the block that you manipulate is highlighted) and the EMACS/Quick... compiler approach, where you move text in and out of a hidden buffer. You can cut text from the context-sensitive help and paste it into the edit buffer. You cannot, however, cut from an arbitrary section of the screen - for example, the DOS output window.

The new system for managing projects - the IDE's equivalent of a MAKE function - is definitely an improvement, although old

Borland hands may have to fight with it a bit at first. Turbo C V2.0 .PRJ files had to be typed in manually as text, and consisted of a list of sources and .OBJS, one per line, to be used in the project. Dependencies had to be placed next to the name of the file that used them. The new system automates all of this. You can just click on the your selection from the list of files and it is added to the file. The IDE finds all the dependencies for you, so *you* can now use *it* to browse through the `#include` files that you have sucked in. Compiler options are stored in the project file; you can specify a global default for compilation switches and file-by-file exceptions. There is a facility to use the 'wired-in' external translators mentioned above, so you can incorporate that module written in Pure Dorset LISP into your C++ project.

The feature that may confuse newcomers is that the setup of desktop windows is stored in the current project file. When you switch projects, all your windows suddenly close themselves down, then reopen somewhere else with different contents. You can also store textual notes in the project file but, as you will have deduced, it has a binary format, so can only be altered by the IDE. Utilities are provided to convert Turbo C .PRJ files to Turbo C++, and also to generate conventional MAKE files from TC++ .PRJ files.

The IDE debugger has been fixed up a bit; it now has conditional breakpoints and an inspection facility that handles arrays, structures and classes. Borland has not put too many features into it, however - remember it intends to sell you the separate Turbo

Debugger.

A final note about the IDE. It manages to squeeze a lot of code into the MS-DOS environment (it does contain a C++ compiler, after all) and yet can still compile reasonably sized code modules. How does it manage this? By making extensive use of Borland's overlay system VROOMM (Virtual Runtime Object-Oriented Memory Manager). You can generate your own VROOMM applications with this package, so a more detailed description is in order. Modules (called 'segments') to be overlaid are compiled with a special switch. Stub code is generated for all the segment's entry points. When the program is run, calls to the stubs go to an overlay manager, which determines whether the required segment is currently loaded. If it is, the call is passed on. If not, the segment is loaded in from disk. If there is insufficient pool memory to allow it to be loaded, discard loaded segments. Note that, using this scheme, overlaid functions can call each other, so the setup requires minimum programmer effort to function. Note also that segments may be discarded to expanded or extended memory, permitting very frosty reload times when their turn next comes around.

What you get

The compiler, or rather the *compilers*, for there is a different parsing/front-end mechanism for C and C++, is/are integrated into the environment seamlessly, thanks to VROOMM. I mention this because, if your first experience of an integrated environment was Microsoft C V6.0 PWB running under DOS, you might have gained the impression that they are mostly about staring at the screen while the compiler swaps itself in and out. (Incidentally: MSC6 users, the word on the street is that the cure, ha-ha, is to run PWB under OS/2.) It has been a traditional feature of Borland's languages that small files compile almost instantly. I am pleased to report that this feature has been maintained, and that Turbo C++ is about the fastest (C) compiler around, although it is not as fast as Turbo C V2.0.

There are two gains to be had from waiting longer for your object code to be delivered: thoroughness of language parsing and quality of object code. The compiler is now better at picking up obscure errors than it used to be. For example, I ran TC++ over a version of the Tetris game that I wrote for Turbo C V2.0. It spat out an error where I was trying to write to a dereferenced pointer to a structure. On closer investigation, it turned out that I had declared a pointer to a constant structure, instead of a constant pointer to a structure (stop snig-

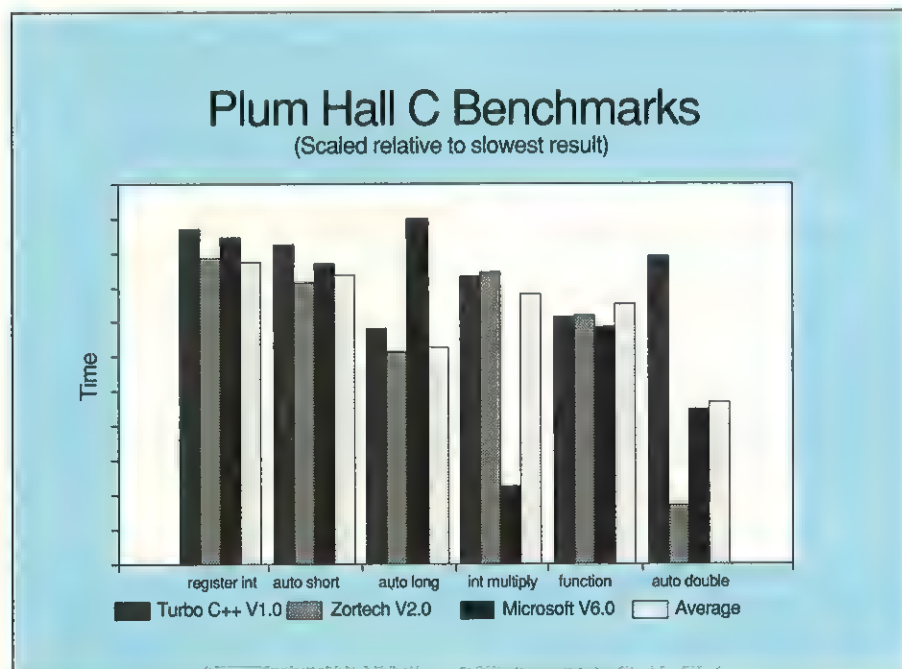


Figure 2 - Graph of Plum Hall C Benchmarks

GENUS SYSTEMS

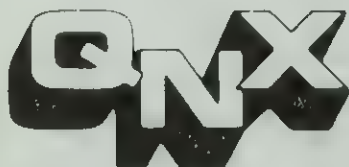
TECHNICAL AND SYSTEMS SOFTWARE SPECIALISTS

Distribution of complete ranges of carefully selected quality software products including:-

Lattice

THE only C development environment to support everything from a 640Kb PC/XT to a 16MB 80286/80386 using MS-DOS or OS/2. A proprietary MS-DOS extender matched to the compiler is included in the package and permits very large programs to be run under MS-DOS in a straightforward fashion. The CodePRobe debugger, operating in protect mode, speeds debugging by preventing a program corrupting itself. Lattice C generates very highly optimised code, coming top in PC Weeks recent test of OS/2 C compilers

Prices from £175



THE only truly distributed operating system for the PC and PS/2 class computers. Contact Genus for free system configuration and sizing.

Prices from £395



THE industry-standard range of Transputer compilers available for both MS-DOS and also QNX via Genus' TransputerNet environment. This latter permits sharing of Transputer cards around a QNX network.

Prices from £750

Services offered include:-

Systems design/sizing. Technical consultancy. Ada/Pascal/C/Fortran /Assembler programming. Courses on all products distributed.

Further info, demo disks etc from:-

Genus Systems Ltd, 9 Waters Close, Leith, Edinburgh EH6 6RB

Tel: 031-554 0455

Fax: 031-554 0456

SEE US IN MUNICH AT SYSTEC 90 WITH THE BRITISH GROUP

gering, it could easily happen to you), but that Turbo C V2.0 had happily allowed me to go ahead and walk all over my supposedly protected data. In this case it did not matter, but you can easily imagine a situation where a safety feature like this would be a life saver.

This tighter checking of the source follows from the requirements of C++, which is less tolerant of monkeying around with types than its parent language, and the requirement of ANSI conformance. This has come a long way since Turbo C V2.0, which failed 16 out of 79 tests in the famous Plum Hall ANSI conformance test suite (described by the BSI's Neil Martin in PCW April '89). Turbo C++ scores 73 by itself and, if you pass the test code through a supplied utility called TRIGRAPH.EXE, it scores 78. (Borland takes the view, correctly in my opinion, that trigraph sequences, such as `??=` for `#`, are so rarely used, there is no point in wasting the time of the ordinary punter by processing them in the main compiler itself. Apparently, omitting the ability to cope with trigraph sequences speeds things up significantly.) The failure is in the test to cope with multi-character *character* constants, such as `'1234'`. This places Turbo C++ up there with the ANSIest of compilers, the others being JPI TopSpeed C V1.0, Lattice C V6.0, Microsoft V6.0 and Watcom V7.0.

Borland also claims to provide a full implementation of AT&T's C++ V2.0 specification - ie the version that demands multiple inheritance and pointers to members (the latter is missing from Zortech V2.0). The company says that it will track new releases, such as the recent V2.1 (which is not much different from V2.0) and the forthcoming ANSI version. Obsolete features, such as the keyword `overload`, which was previously required when overloading function declarations, can be used, and generate a suitable warning message. But it is over another V2.0 feature, type-safe linkage, that I have a quarrel with Borland's design.

Mangled

AT&T's standard requires that inter-module linkages to functions take account of the function's signature as well as its name. In other words, under C++ you must not be able to refer to a function defined in one module by different prototype (with the same function name) supplied in another. This type-safe linkage is achieved by inventing a new name for the function, based on the type and number of its arguments plus, if it is a method, its class. The Specification says nothing about how this mangling should be achieved. However, there is a bundle of papers that are distributed with the specification as 'Selected Readings'.

One paper, by Bjarne Stroustrup himself, is called *Type-safe linkage for C++*, and it includes an algorithm for this mangling. This is the algorithm used by AT&T's cfront compiler (and hence by Glockenspiel's compiler), and adopted by Zortech for its compiler. Borland, on the other hand, uses a little something of its own. Given the method

```
intobj1::fn1(inti,intj)
Zortech produces _fn1__4obj1Nii,
Borland puts out @obj1@fn1$qi.
```

The problem, of course, is that a class library compiled for use with Borland's compiler won't work with anybody else's, and vice versa. Did Borland choose to make its product incompatible for marketing reasons, or out of incompetence, because the development team didn't spot Stroustrup's paper until too late?

A glance through the results of the various tests that I have done (which, although unrepresentative, unrealistic etc, confirm my subjective observations) will show you that, as an optimising compiler, Turbo C++ is nothing to write home about. The timings, incidentally, were made with the command line version of the compiler (oh, did I not tell you there was a separate command line version? Well there is, along with a linker, a librarian and a conventional MAKE utility). It has noticeably improved from the previous release - it seems to make better use of registers than Turbo C V2.0 - but it is not up there in the ozone with the Watcoms of this world. However, as Adam Denning noted last month in his review of Microsoft C V6.0, a man who buys a compiler on good performance alone is a fool. There again, a man whose code runs faster may sell a few more copies of his package...

Libs and Docs

Turbo C++'s libraries are very similar to those of its predecessor. There are still the dear old BGI routines for handling graphics. The exotic ANSI functions prototyped in `LOCALE.H` have been added. There are a few classes: complex numbers, supported by overloaded trigonometric functions, BCD, a set of generic data structures, such as lists and queues (these are all derived from a single object type, Smalltalk style, and are documented only on disk, for some reason) and I/O streams, old and new (V2.0) style.

The inclusion of new-style streams is one over Zortech's effort, which only offers the old sort. Whether this matters to you depends on whether you belong to the 'cout is brill, it's just like Pascal's write, you can forget about types' camp, or the

<pre>;Turbo C V2.0 ;Time per iteration in ms= 40.94 _sieve proc near push bp mov bp,sp sub sp,8196 push si push di xor di,di jmp short @5 @4: mov byte ptr [bp+di-8192],1 inc di @5: cmp di,8190 jle @4 mov word ptr [bp-8194],0 xor di,di jmp short @9 @8: cmp byte ptr [bp+di-8192],0 je @7 mov ax,di add ax,di add ax,3 mov word ptr [bp-8196],ax mov si,di add si,ax jmp short @11 @13: mov byte ptr [bp+si-8192],0 add si,word ptr [bp-8196] @11: cmp si,8190 jle @13 mov ax,word ptr [bp-8194] inc ax mov word ptr [bp-8194],ax @7: inc di @9: cmp di,8190 jle @8 mov ax,word ptr [bp-8194] pop di pop si leave ret sieve endp</pre>	<pre>;Turbo C++ V1.0 ;Time per iteration in ms= 36.63 _sieve proc near push bp mov bp,sp sub sp,8192 push si push di xor di,di jmp short @1098 @1050: mov byte ptr [bp+di-8192],1 inc di @1098: cmp di,8190 jle short @1050 xor bx,bx xor di,di jmp short @10290 @10146: cmp byte ptr [bp+di-8192],0 je short @10266 mov ax,di add ax,di add ax,3 mov dx,ax mov ax,di add ax,dx mov si,ax jmp short @10218 @10194: mov byte ptr [bp+si-8192],0 add si,dx @10218: cmp si,8190 jle short @10194 mov ax,bx inc ax mov bx,ax @10266: inc di @10290: cmp di,8190 jle short @10146 mov ax,bx pop di pop si leave ret _sieve endp</pre>
---	---

Figure 3 - Object code for Sieve function

PANEL[®] Plus II



Screen Generator and User Interface Library for 'C'

PANEL Plus II features: Interactive screen design editor
• C and MS-FORTRAN code generators • library of user-interface functions • pop-up fields/windows • multi-line fields • mouse support • scrolling regions/scroll bars • help boxes • pull-down menus • custom validation • shadow borders • complete **library source**, with low-level source and headers for all supported systems (DOS, OS/2, Unix, VMS) • interfaces to popular PC **graphics libraries** • OS/2 DLL • **no royalties**.

Also available: **Utility Source Option** which allows **PANEL Plus II** to be ported to any suitable development platform. *New in version 2.1:* **CUA/SAA** compatible key assignments and edit functions, CUA 'function key area' support, word-wrap editing, new menu styles, **MS-C 6.0** interface (with on-line help), DOS-extender interfaces.

Panel Plus II licence fee per programmer: **£295** (including library source and 700-page manual). With utility source option: **£720**. Please call for quantity pricing, full version list, and free demo disk.

Roundhill Computer Systems Limited

Orchard House, Ogbourne St. George, Marlborough, Wiltshire SN8 1SU

Telephone: **0672 84 535** • Fax: **0672 84 525**

Roundhill also supplies compilers and development tools from Copia International, Graphic Software Systems, Lattice, Periscope, Pocket Soft, Polytron/Sage and others. Please call for a full catalogue and price list.



Not Entirely Unbiased

Steve Teale is a man who knows all about MS-DOS C++ compilers: he writes Zortech's class libraries. In an impish mood, we asked him what he thought of Turbo C++.

A couple of months ago, my association with Zortech being known, I jokingly asked Will Watts whether he would like me to do a partisan mini-review of the Borland C++ compiler when it came out. That will teach me!

In the event, the Borland release date left a tight publication deadline, so I scratched my head to think of a some code which would tax the two compilers, and decided to try first an implementation of extended two's complement integer arithmetic, with overloaded operators, lots of implicit constructor calls etc.

It's an interesting property of different compilers that, given the same code, they will produce different sets of error messages. If code has been developed with one compiler, another will generally throw a different light on it. Needless to say my own code was not immune in that respect. This fact of life doesn't favour a compiler with an existing body of code, compared to a new compiler with hardly any code to its name, but any critics should remember that in the longer term such traffic is two way!

The appearance of a C++ compiler from one of the bigger guns of the DOS world might give cause for thought to C programmers who, until now, have got by thinking that C++ is just a gimmick. Maybe it is time to try. But if you want to develop programs for Windows 3 and Presentation Manager, then the Borland release will be a non-starter for you, since it doesn't handle these at all.

Another choice you will have to make is over the split which seems to have developed in the C++ world. On one hand there is what might be called the AT&T camp. C++ is AT&T's baby, and they have consistently emphasised C++ as a better C, a general purpose programming language with facilities for object-oriented programming. On the other hand are the pure OOP devotees, who really wish that C++ was Smalltalk.

A look at the class libraries provided as part of the Borland C++ package seems to indicate that Borland belong in the latter camp. If your view is 'and so they should', I suggest you take in the 'Silver Bullet' article by Niklaus Wirth in .EXE Magazine, May 1990. This piece takes a long, cool look at some object-oriented excesses. In saying this, I am not putting down OOP. It is great in its place, but silver bullets are scarce. Zortech has tended to adopt a more conservative approach.

Enough, back to my ISOLATED test. The program does a million different long multiplications and divisions. Both compilers produced the same result, but with interesting performance differences, as you can see in Figure 1.

This was not a contrived example, but a piece of production code which I chose because it was almost library independant, and not I/O bound. Borland clearly have some way to go, and Zortech is not a stationary target!

	Compile time	(straight compile)		(compile for speed)	
		run time	.EXE size	run time	.EXE size
Borland	6.10	748	29658	755	29722
Zortech	5.54	403	25116	385	24556

Steve Teale develops C++ class libraries up in Harrogate, where he finds time to work on his own consultancy, Microcosm Computing (0423 562055).

The code that Steve used for his test forms part of Zortech's forthcoming C++ Tools, so he is unable to place it in the general public domain. However, he has made it available to us, so we can verify that there was no monkey business! It is our intention to invite a Borland representative to write a 'return match' article when the new Zortech compiler, due for release in early July, becomes available.

'printf()' may be old-fashioned, but at least you can see what it is doing' camp. The function documentation is bountiful, included both on disk and paper, and has been expanded. For example, the description of the DOS Keep() routine includes the listing of a small TSR program which hooks into the clock interrupt.

Other documentation also maintains the traditional, high Borland standard: there is no attempt here to fob you off with a quick reference card and some hypertext help files. There are four fair-sized perfect-bound manuals, entitled *Getting Started* (which includes C and C++ language tutorials), *User's Guide*, *Programmer's Guide* (contains ANSI mandatory implementation-specific features list) and *Reference Guide*. One complaint. There is no explanation of the name-mangling algorithm, not even in the mixed language programming section.

Conclusion

Borland is one of the big boys in the MS-DOS compiler world; the fact that it has a C++ compiler is significant, because it improves the chances of C++ gaining general acceptance. This, however, is a marketing observation. In .EXE, we like to retain a technical bias. Technically speaking, Turbo C++ is a bit disappointing.

TC++'s output code is lacklustre; its selection of class libraries, albeit inclusive of AT&T's V2.0 stream I/O, is not large. TC++ is binary incompatible with libraries for Zortech, and probably any other PC compiler that happens to come along. There is no support for Windows (although this is promised within a few months) or OS/2 (this is just promised). Plus points: there is a good system of overlays, the integrated environment is certainly cute, you can use Turbo Debugger on the programs that you produce.

During my career as a programmer, none of the software tools that I used were as slick or as neat a Borland's: this is how I became a fan. Turbo C++ seems to lack the special sparkle of the earlier efforts; it is steady rather than brilliant. I hope that they remember to put back the sparkle for next time.

[EXE]

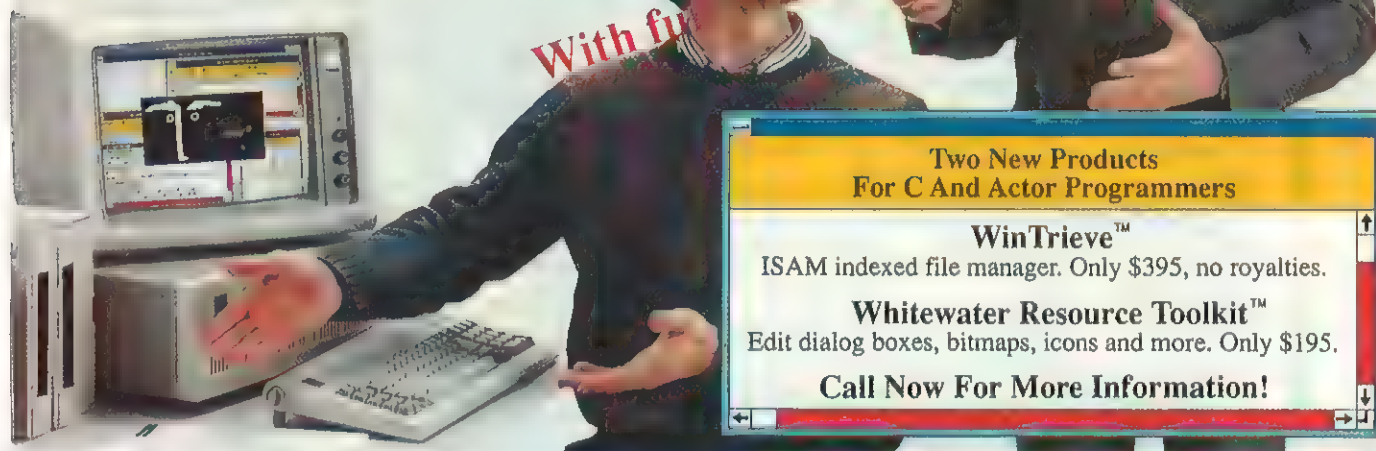
Turbo C++ V1.0 retails at £149.95, Turbo C++ Professional (with extra programmer's tools) costs £249.95. The software is available from dealers. There are special deals available for registered Borland users (not just Turbo C) - the company will contact them soon with details.

Develop Microsoft® Windows Applications Fast!

"How Did You Get
This Windows Application
Done So Fast?"

"Everyone Else Is Still
Reading the Manuals..."

With full Windows 3.0
support



The Windows Advantage

You've selected Microsoft Windows as your graphical user interface. Windows applications are easier for end-users to learn and use. It is the standard for PCs.

Now you want Windows applications completed as fast as possible. But it takes time for your programmers to get up to speed on the intricacies of Windows development.

Why The Delay For Windows Applications?

Programming for the Windows environment is very different from character-based programming. Just learning the 450 Windows function calls can take months. Getting used to its event-driven system and memory management takes additional time.

Programmers who are new to Windows development are looking for support to help them produce high-quality Windows programs quickly and efficiently.

3 Ways To Speed Up Windows Application Development

Now there are specific ways to speed up applications development under Windows.

- **Code reduction.** There is a fast, effective way to develop Windows applications. A way that cuts the amount of

code needed. A way that by-passes much of the complexity of Windows programming. A way that doesn't compromise performance.

- **Short learning curve.** There is an easy, convenient way to get your entire programming staff up to speed with Windows within weeks.
- **From prototype to complete application.** There is a way to quickly develop prototypes of Windows applications in days, not months. Then, turn those into completed applications quickly.

1 Product, 3 Solutions

There is **one** product which brings **all three** solutions to your Windows programmers. It's called **Actor**®.

Actor is a proven product. It's now being used in thousands of software development groups world-wide.

Actor Does Windows

Actor gives your programmers a fast, effective way to develop Windows applications. **Actor's** complete development environment allows you to produce everything from prototypes through stand-alone applications. Programs developed in **Actor** are fast and support all Windows features.

Actor offers interactive source code debugging and testing—greatly reducing development time. Your programmers

benefit from high code reusability because **Actor** is an object-oriented programming language. This allows them to build on existing code, eliminating duplication of effort. They can even dynamically link to existing C code.

Actor also provides one more essential: fast, responsive support from The Whitewater Group®, recognized worldwide as leaders in Windows programming.

If you are programming applications for Windows, you should be using **Actor**. Users report that **Actor** cuts Windows programming time by at least 50%.

Act Now!

See for yourself how **Actor** can help your Windows programmers! The first step is to order our free booklet, **For Faster, Better, Smaller, Smarter Windows Programming**. Examine detailed information about using **Actor**, complete with actual source code examples.

Call us at 312/328-3800. We'll send you the booklet and complete details about **Actor**.

Take 60 seconds now to save months of development time in the future.

NEOW - The Windows Specialists

Unit 12, Progress Business Centre,
Whittle Parkway, Slough, SL1 6DQ.
Telephone: 06286 68334. Fax: 06286 65722

All trademarks acknowledged

CIRCLE NO. 071

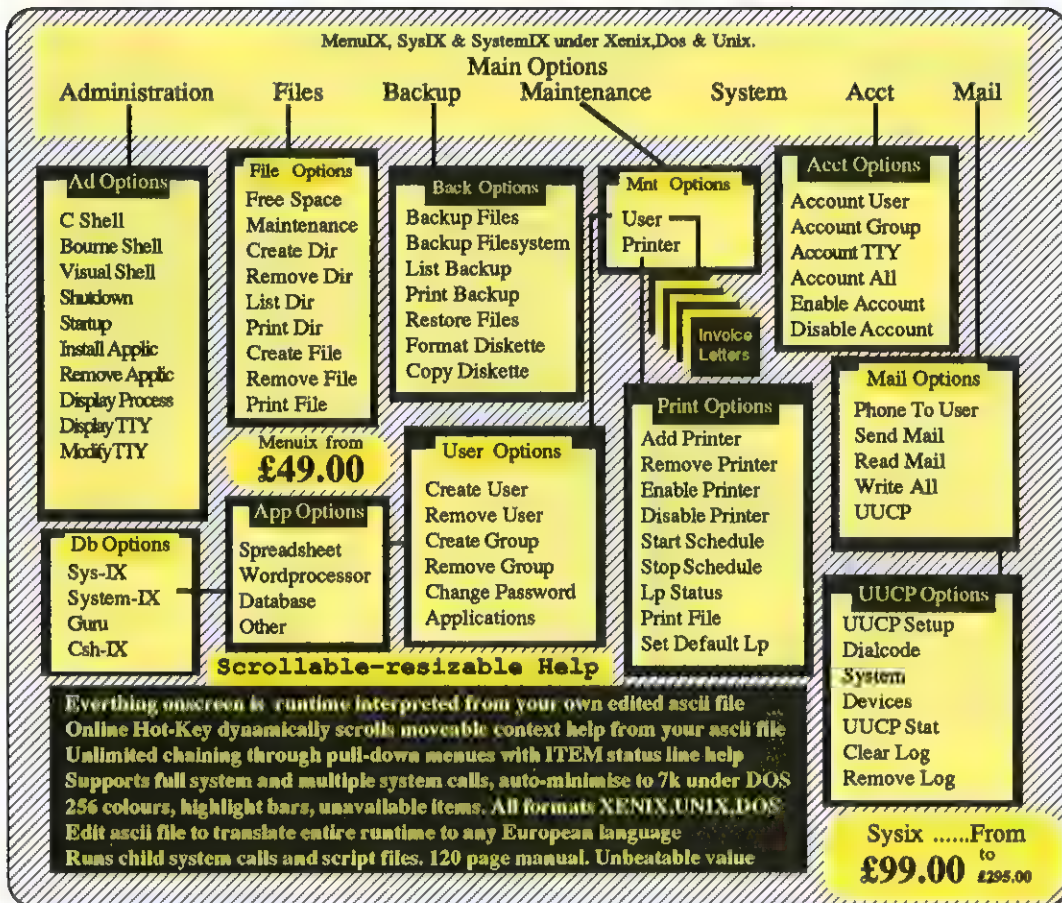
G.W. Computers Inc.
ADCL 4 Eagle Square
E. Boston, MA 02128
U.S.A.

Tel: **617-569-5990**
Fax: **617-567-2981**

System-IX (Networks) Ltd.
55, Bedford Court Mansion
Bedford Avenue, London
WC1B 3AD U.K.

Tel: **071-636-8210**
Fax: **071-255-1038**

Menu-IX System-IX DBMS-IX Sysix AI Spell Merge Lock GURU Crypt Tawk Tgrep



Powerful Data Entry

Sys - IX data entry is easy to use for the new user and powerful for the experienced professional. A multi-level design gives high level functions ease of use leaving lower level functions the versatility you will demand.

This makes Sys IX perfect for users of all levels of expertise for any application.

Input formatting with pictures, Complete colour control for inputs, Powerful validations, Pick from selection sets, Definable input control keys, Security level for each field or screen. Different status line for each field, Right or left numeric input, Floating dollar signs. Commas inserted into numeric fields, Insert or overstrike mode, Pre-function or post-function for any field, Field specific help messages, Automatic scrolling fields. Abort screen without changes, Option for automatic carriage return, Optional field blanking on first keystroke. program accessible for customising, Single field or full screen input, Move field next, previous, up, or down. No. fields/sr limited only by RAM, Input masking, Change one field based on another. Skip fields.

Fast Windows

Window management is totally automatic with Sys - IX. After a window is opened Sys - IX will allow you to print to any window, at any time, without problems. No structures to fill, no questions to answer - just open a window and all output will be directed to that window - it's just that simple.

Automatic collision protection, Printing to any window, Bordered or borderless windows. Runtime adjustable help windows, Number limited only by RAM, Move or reshape windows under program control. Interfaces to all data entry windows, Zoom in or zoom out windows, Pull-down windows. Vertical or horizontal scrolling, Choice of borders, Pop up Windows. Virtual screens automatically scroll, Advanced screen optimization, Unix extensive Termcap file for extra features. Supports all terminals, Supports colour, Supports line graphics for windows, lines, and boxes plus optional scroll bars. Extremely portable, Auto. restricts I/O in window. Auto-select colours or manual, Print to background and foreground window at the same time.

Includes POWERFUL Data Transfer & Reporting Utilities on full system

System-IX INCLUDES: GURU, CRYPT, AI, BTGURU, PLUS SEVERAL APPLICATION PACKAGES.

UNDER DOS, also includes: CShell, AWK, GREP, FOREP, DIFF, CAL, HEAD, TAIL, MORE, CAT,

TOUCH, TEE, SORT, FIND, DETAB, ENTAB, ENCRYPT, DECRYPT: (Subject to variation without commitment).

Compaq Toshiba IBM Epson Qume NEC Seagate Wyse Relysis Zenith Primar Ranger

Program Structures of the Fourth Kind

The search for structure in computer programs, especially structure avoiding the use of GO TO, has been going on for over 20 years. Richard Pickard offers a critique of the options in different programming languages, and suggests an alternative.

During the 1950s and 1960s, the 'bad old days', a succession of procedural programming languages was developed, each at a 'higher level' than its predecessor. For most of this period, the bulk of the design effort was aimed at making it easier and faster to write programs. There was little to improve the quality of code: the new high-level programs didn't work in just the same way that the old assembler and machine-

code programs didn't work. Because the compiler writers' thinking remained heavily oriented towards the machine and its features, high-level languages commonly offered only a limited range of data types and simplistic control structures.

From the machine's perspective, 'data processing' was mainly about 'processing'. 'Processing' meant doing calculations,

moves and tests; and looping, branching and calling subroutines. 'Data' was in the province of the application - nothing to do with the tool being used - and compilers did little or nothing to check the legitimacy of what operations on data were being attempted. Certainly, the languages of that era provided a variety of data types for the convenience of the programmer, but these types were predefined - even in COBOL - and only rarely were there any strong rules about the ways in which they could be combined.

However, it is structure issues that concern us here. The main point about structure - as it was first offered in high-level languages - was that it was barely recognised to exist. The original IF statement in FORTRAN was merely a three-way conditional branch; COBOL did a little better with the 'conditional sentence'; not until ALGOL did we have a clear and usable way of nesting IF's. The situation was a lot better with loops; these, by definition, have a clear scope. (The scope is specially clear if, as in COBOL, the loop body is out of line and referred to by its label.) All early languages provided procedure calls and returns - most of them allowed a return to the calling procedure from any point in the body of the called procedure.

Going, going, gone

The weakest area was probably that of interrupt and error handling. It was hard to find a high-level language providing direct control over hardware interrupts and software events with equal power. The handling of hardware was (and is) not an integral part of most so-called system programming languages - it was provided by libraries of assembler routines. Language support for the handling of unpredictable software events was similarly missing.

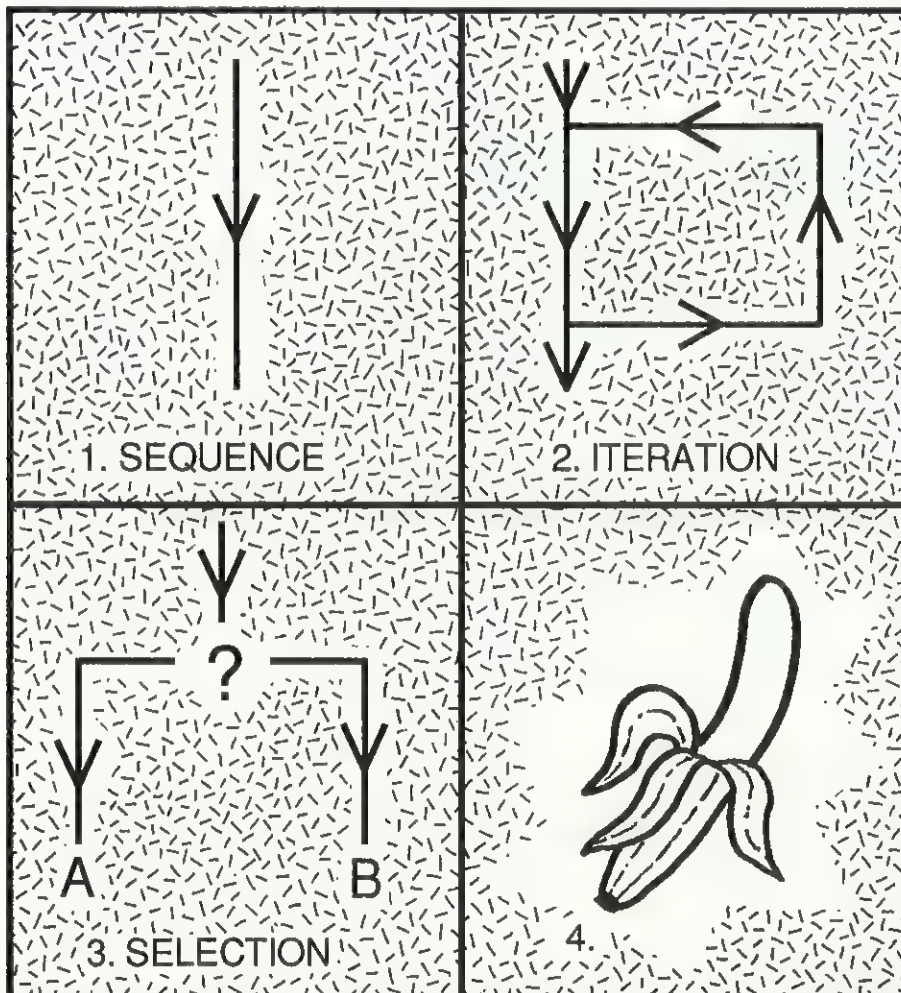


Figure 4 - The same solution expressed with the POSIT

```
...
POSIT NoFataIs:
  POSIT FileWillEnd:
    LOOP
      MyReadNext (ProdFile, ProdRec,
        FileWillEnd, NoFataIs.BadFileStatus) ;
      WrLn ; WrCard (ProdRec.ProdCode, 7) ;
      WrStr (ProdRec.ProdDesc) ;
      POSIT LookUpWillBeOK
        WrStr (SupplierName [
          MyLookUp (Supplier, ProdRec.SuppCode,
            LookUpWillBeOK.NotFound) ;
        ]) ;
      DENIAL NotFound:
        WrStr ("Supplier name not found for code: ") ;
        WrCard (ProdRec.SuppCode, 0) ;
      END LookUpWillBeOK ;
    REPEAT ;
  END FileWillEnd:
  WrLn ; WrLn ; WrStr ("End of list") ;
  Close (ProdFile) ;
  DENIAL BadFileStatus:
    WrLn ; WrLn ; WrStr ("Product file error: ") ;
    WrCard (ProdFile.FileStatus, 0) ;
  END NoFataIs ;
...
```

it allows the drawing of an EXIT from any block to any outer level. Because the mouse is used to point to the outer level, no labels are involved (in a text-only language such a handle cannot be avoided). KnowledgeWare's EXIT is equivalent to the PROVE of the POSIT.

I would like the question of how to replace the GO TO reopened. The generality of the loop structure has been successfully implemented in the PLANC language (*The Third Side*, .EXE November '89) and should be copied by others. In the same way, we need a generalised block-and-escape structure, such as I have described here.

EXE

Dr Richard H Pickard has been a full-time data processing practitioner and writer on the subject since 1964. He is equally interested in the application of DP techniques to business problems as in the techniques themselves. He is now a senior computing specialist for Whitbread. Richard may be contacted on 0525-61836.

Does any language which you know allow this kind of control? The only thing near it which I have seen is the Action Diagram, in KnowledgeWare's Analyst

Workbench. This is a picture-and-text editor for the specification of structured procedures. It allows the drawing of Sequence, IF and WHILE blocks and



Help!

.EXE Magazine is looking for new contributors. In future months, we will be producing special issues with in-depth coverage on the following topics:

- **September** - Pascal and Modula-2
- **October** - Security and software protection
- **November** - Object-oriented programming
- **December** - Debugging techniques

Plus a Real Time supplement with November's issue.

If you have expertise in one of these fields, and would like to write for .EXE Magazine, why not get in touch? Whether you have a clever routine which solves a mind-twisting problem, have brilliant insight into some impenetrable area of academic research, or just have in-depth, practical knowledge of one of these subjects, we should like to hear from you. Please send a 200-300 word synopsis of your proposed article to:

**The Editor .EXE Magazine,
10 Barley Mow Passage, London W4 4PH**

PS - If you have an idea for a brilliant .EXE article, but it does not fall into one of the above categories, write and tell us about it anyway.

PPS - We pay real money for articles that we publish.

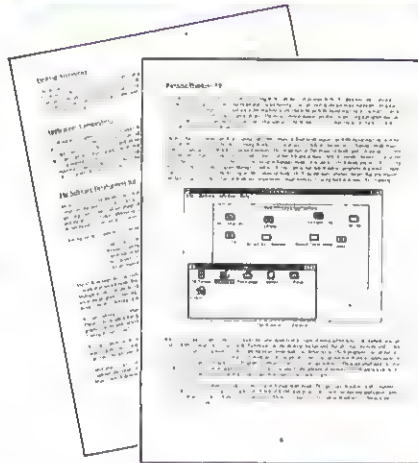
.EXE

Special Offer to .EXE Readers

24-PAGE NEW PRODUCT BRIEFING

.EXE has prepared a thorough evaluation of Windows 3. This includes a full listing of the programming API, plus details on memory management, network operation, protected mode operation and other key details.

If you are considering using Windows 3.0 but you don't know whether OS/2 is a better choice, the New Product Briefing will help you decide.



The Windows 3.0 New Product Briefing costs £14.95, including P&P. To get it fast, call 081 994 6477. Just quote your credit card details or an official company order number, and we'll despatch your copy within 48 hours. Alternatively, send a cheque to *.EXE Magazine* at:

10 Barley Mow Passage,
London W4 4PH,
or fax your details to
081 994 1533.

MULTIPLE INTELLIGENT COM PORTS FOR OS/2

THE IMPAC 8 INTELLIGENT COMMS CARD

- * MICROSOFT COM X COMPATIBLE
- * 8 TO 32 FULL MODEM PORTS
- * REDUCE CPU OVERHEAD FROM 40% TO 1% PER PORT
- * FITS ALL ISA (PC/XT/AT) MACHINES
- * DOES NOT USE SHARED RAM
- * PROVEN FIELD PERFORMANCE
- * U.K. DESIGN AND MANUFACTURE
- * DRIVERS FOR OS/2, UNIX, XENIX

DEALER ENQUIRIES WELCOME

For more information contact:
Second Computer Ltd, Reigate
Telephone: 0737-762200

DOS DRIVER ANNOUNCED

CIRCLE NO. 073

FORTRAN

SPAG – Software Renewal

SPAG rejuvenates old FORTRAN programs. It analyses the bewildering jumble of GOTOs, arithmetic IFs etc. which are characteristic of old-style FORTRAN, and re-writes programs using modern structured programming constructs.

Output from SPAG is neatly indented and structured but, amazingly, is exactly equivalent logically to the input spaghetti code.

"a MUST-HAVE product"
(Programmer's Journal, June 89).

Forcheck – Static Analysis

Spring clean your FORTRAN with Forcheck, the ultimate static analyser! Forcheck identifies clutter (unused variables, COMMON blocks etc.), and checks arguments lists and COMMON blocks for consistency. It also produces valuable documentation (CALL trees, cross reference charts etc.), and there is a comprehensive, but optional ANSI conformance check.

SPAG and Forcheck are available for PC, Sun, Vax, Prime, etc.

FTN77/386 – Power User's Compiler

FTN77/386 uses DOS extender technology to turn a 386 based PC into a powerful engineering workstation. With FTN77/386, your programs run under MS-DOS, but have a 4 GIGABYTE address space and super-mini performance. FTN77/386 is a full ANSI 77 compiler validated by the US Dept of Commerce, with many extensions for mainframe and DOS compatibility. A window based source level debugger and profiler is included. Compilation is blazingly fast, and compile and run-time diagnostics are the best available. If you use FORTRAN, you MUST check this one out!

SIMPLEPLOT, the powerful mainframe graphics package, is now available for use with FTN77/386. With SIMPLEPLOT, you can create graphs, charts, contour maps etc. by a few simple calls. For lower level graphics, try our

interface to the excellent
MetaWINDOW
package.

Polyhedron Software Ltd.,
Magdalen House, 98 Abingdon Road,
Standlake, Witney OX8 7RN, England.
Tel: (0865) 300579. Fax: (0865) 300232.

CIRCLE NO. 074



a BREAK with an optional number - to specify how many levels of block from which to escape.

Pascal is the most famous language to be designed specifically for structured programming; curiously, it retains the GO TO and fails to provide a BREAK or a RETURN - each block of code is entered at its beginnings and exited at its end. As Kernighan pointed out (and anyone who's tried it knows) this combination makes for some tricky coding, especially in loops.

The Problem

Now it's time to ask 'what is the real problem?'. We must certainly agree that the writing of easily understood programs in self-contained sections is desirable. It promotes easier debugging and bug pre-empting walk-throughs. It may lead to easier proofs of correctness. These factors taken together raise quality and simplify the task of managing software production and predicting costs.

The problem is that, when we write real programs, it is nearly always the case that we have to cater for errors and exceptions whose occurrence, by definition, must radi-

cally change the course of a program. Consider the simple problem of reading lines of

Testing a flag means that you missed the chance to act when the event occurred

text until end of file. In old-fashioned programming, the end-of-file case would have been handled by a (smelly but effective) GO TO. Some of the 'commercial' programming languages offered escape techniques. PL/1 (circa 1965) provided the ON statement to specify action to be taken if certain software or hardware conditions arose; these conditions included both arithmetic and programmer-defined exceptions. COBOL provides the DECLARATIVES sections to trap file- and record-handling exceptions and errors. Some of the newer versions of C (circa

10 years on) and Ada (20 years) have also implemented similar handlers.

But the notion that a block of code could have a natural escape route was not general. The exception-handlers of PL/1 and COBOL were always written before (ie textually above) the points in the program where their conditions could arise; to implement a change of course in the main body of the program, these handlers had to end with a GO TO, or set a variable (flag) to be tested by the main body when it was resumed. There still wasn't a way to express a programming idea such as *Read the next text-line, but if at end of file, get on with the closing sequence (and then the rest of the program)* except to express it with an explicit or implicit GO TO. (I mention the implicit GO TO because of the example provided by the FORTRAN READ statement's optional clause: END=label.) Notice that COBOL's AT END clause doesn't get us out of any holes either.

Escapes are what GO TOs are properly used for. They lead to a program section which deals with some situation which is:

- Radically different from what the program is doing at the point from which the escape is made.
- Common to different points in the program (but not necessarily).

'Now', you may say, 'this is an obvious case for invoking a procedure.' Not so. You only shift the problem to what to do when the procedure returns, just like the ON in PL/1 or the DECLARATIVE in COBOL. This is definitely a job for GO TO.

An area of program construction which has worried me for a long time (as a result of avoiding GO TOs) is the *Why am I here, and where do I go next?* problem. On exit from a loop (never mind how), such as that contained in a table lookup, do we have a valid result? When a READ fails, and the cause has to be unravelled, what has to happen for the different cases of 'No record found', 'File not opened', 'Abort', 'Retry' and 'Fail'?

One technique for handling this problem is to invent an auxiliary variable (flag), whose value is set inside the block doing the work, which is tested on exit from the block. IF or CASE constructs can be used for the test, and the flag has no other use in the program other than this steering function. Sometimes we cheat, and use the result obtained by the inner block in two ways. For example, in the case of a table lookup, the result can be given an 'impossible' value to indicate the failure of the search. Note that this problem and its solution are the same if the attempted

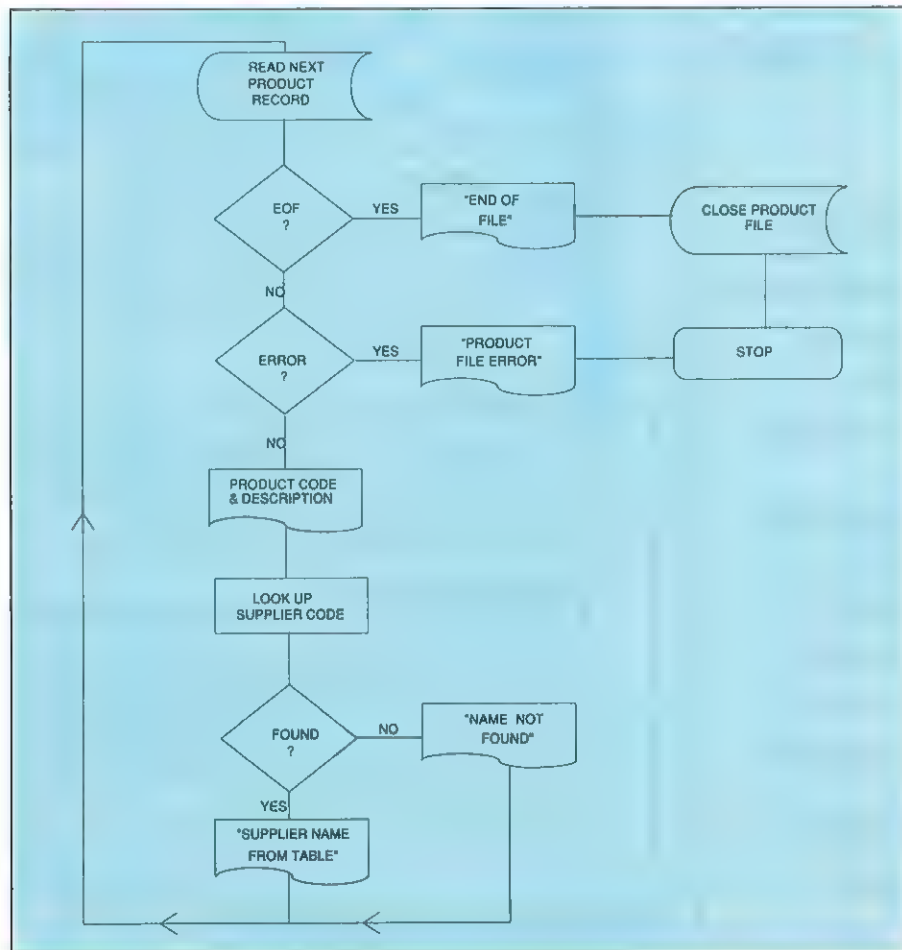


Figure 2 - The sample problem as a flow chart

The Evaluator

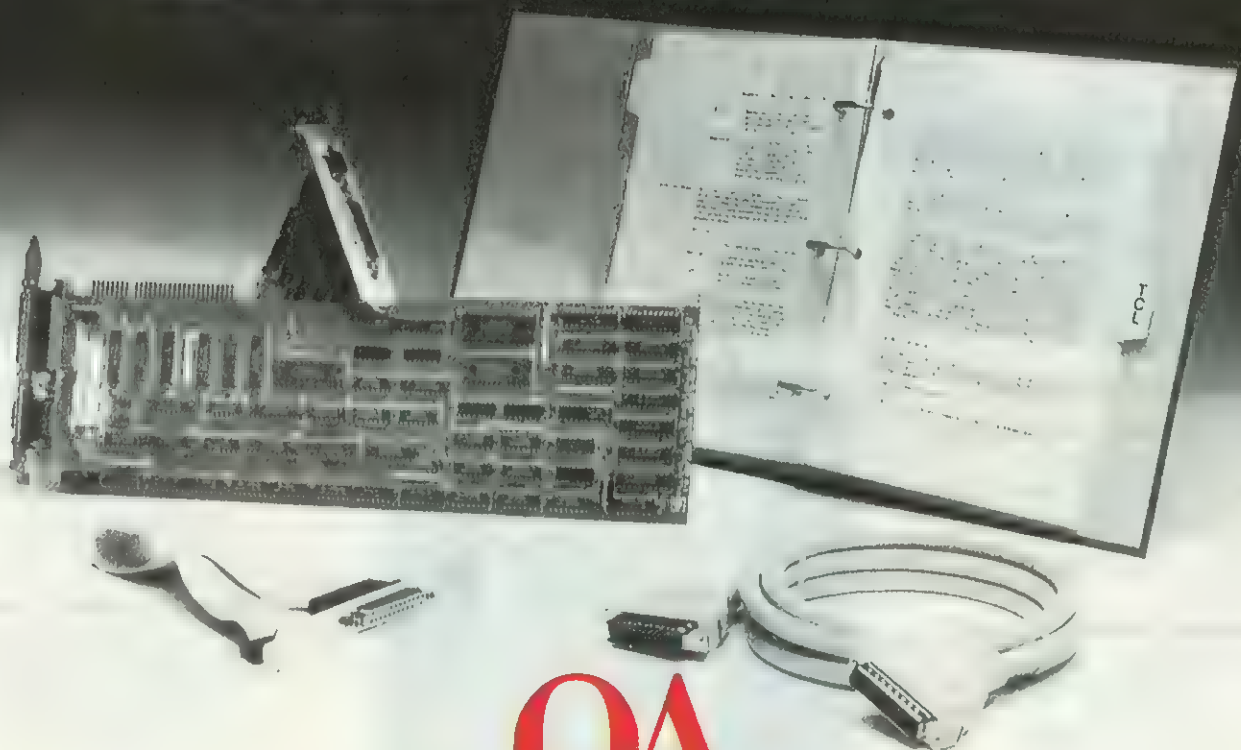
The Breakthrough in Software Testing

Completing the writing of your software is half the battle.

Ensuring that you have produced reliable fault free code is yet another major challenge. Testing software to be certain that full functionality is met during development or maintenance cycles is costly, time-consuming and prone to error...until now.

Evaluator, created by Elverex and supplied by QA, both simplifies and automates software testing, allowing you to create a comprehensive test process which fully exercises software to detect any errors. It saves effort, repetitive work, and frees the time of test personnel.

- Interactive recording and playback of complex test sequences
- Rich test control language to write test programs
- Totally non-intrusive testing of any application on any operating system - DOS, Unix, OS/2, VMS, RMX...
- Automatic test report generation
- Unattended operation allows for more productive use of test personnel
- Easy to use control software for test management and monitoring
- Uses standard IBM PC or AT or 100% compatible as test station
- Tests software that runs on terminals or on a PC

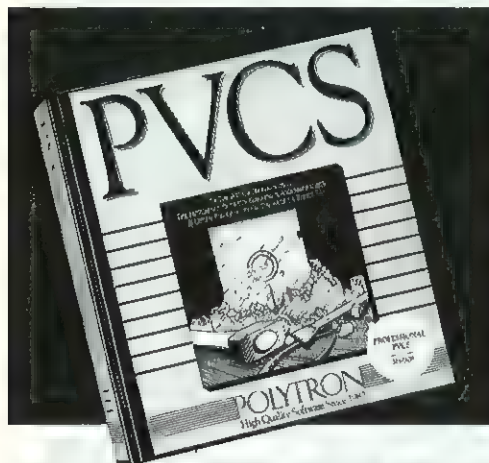


QA

Telephone us on (0285) 655888 for full information. QA, Cecily Hill Castle, Cirencester, Glos GL7 2EF

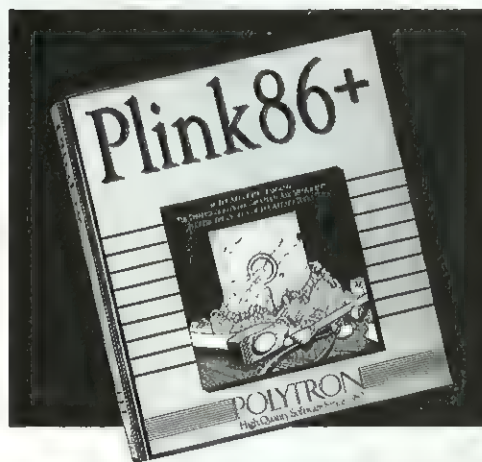
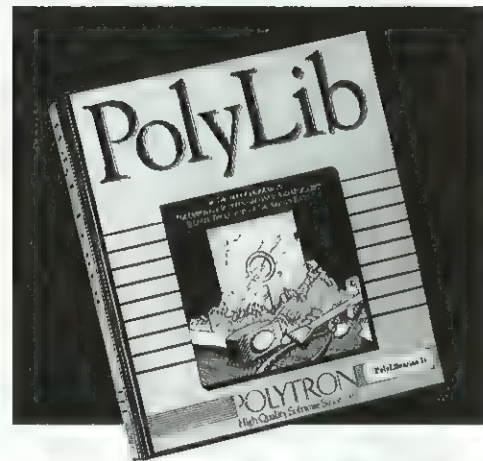
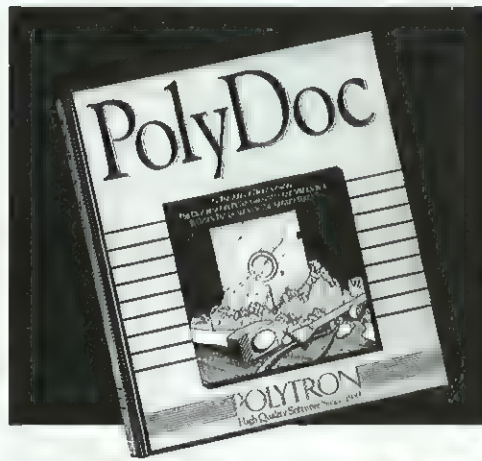
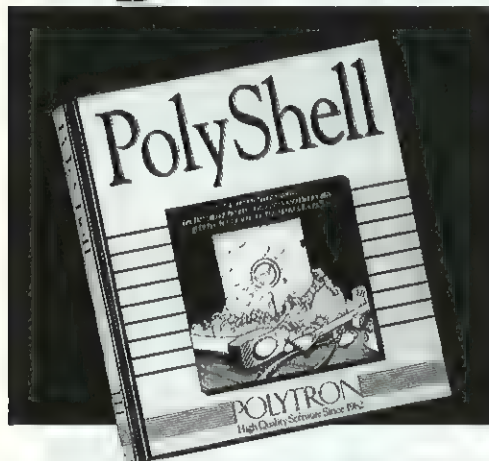
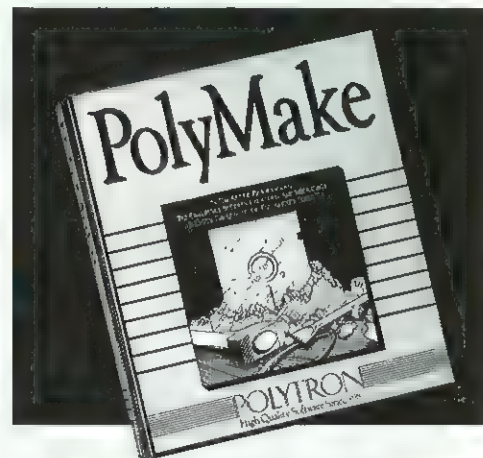
POLYTRON

SOFTWARE DEVELOPMENT TOOLS



FROM

 SOFTWARE
 GENERATION
 SOLE DISTRIBUTOR
 0462 422525

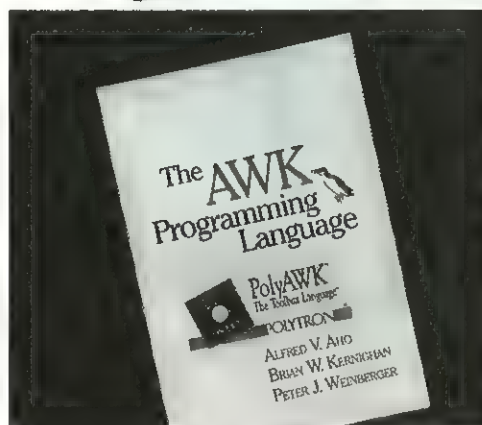


APPROVED DEALERS

Roundhill Computer Systems Limited
 Orchard House
 Ogbourne St. George, Marlborough
 Wiltshire SN8 1SU
 Tel: (0672) 84535

Computer Solutions Limited
 Canada Road, Byfleet
 Surrey KT14 7HQ
 Tel: (09323) 52744

Grey Matter Limited
 Prigg Meadow, Ashburton
 Devon TQ13 7DF
 Tel: (0364) 53499



Software Paradise
 Avenue House King Edward Avenue
 Caerphilly Mid Glamorgan CF8 1HE
 Tel: (0222) 887521
 CIRCLE NO. 077

APPROVED DEALERS

Frontline Distribution Limited
 Intec 1, Wade Road, Basingstoke
 Hampshire RG24 0NE
 Tel: (0256) 463344

Reflex Technology Limited
 9 Buckingham Place
 Bellfield Road, High Wycombe
 Bucks HP13 5HW
 Tel: (0494) 465907

System Science
 3-5 Cynthia Street
 London N1 9JF
 Tel: 071-833 1022



work is done by in-line code or by a separate procedure.

When procedures are indeed written separately, they can be constructed as functions, and substituted in the places where an auxiliary variable might have been used. This saves the program itself having to declare the variable, but it will probably still exist, hidden inside the function's definition.

The Solution

I would like to reduce programs to their essentials. If a section of code can have different successor sections, depending on the conditions encountered during the execution of that section, then - right at the point where the condition is identified - I would like to arrange to execute the relevant successor. If a variable must be created just to convey the significance of some earlier result, and will only be used once, to determine the next action then I don't want it. Having to test a flag means that you missed the chance to act when the event occurred.

We are looking for a control structure which satisfies the following requirements:

- To escape from the current block to some later block.
- To escape up through any number of levels.
- To arrive at the destination with some idea of why the escape was made.
- To retain a block structure in which uncontrolled branches cannot be made (accidentally or deliberately).

The solution - which is not wholly my own, and has its roots in academic work done in 1970 - is to use a new kind of

When they threw out the GO TO, maybe they were too quick to throw out the label

block and a restricted form of GO TO. This new block - the fourth kind - is called a POSIT ('let it be supposed'). As you can see from Figure 1, it has a main body and any number of conditional parts. The POSIT and the conditional parts are all given names. Normal program flow is from the beginning to the end of the main body and then to the next block at the same level, ie after the last conditional part. Abnormal flow is initially from the beginning of the main body and - at some point where an exception is detected - to just one of the conditional parts and then on to the next block. The way of making escape is with a PROVE statement, which nominates the POSIT, or with a DENY statement, which nominates the POSIT plus a conditional part. PROVE makes a

complete escape, directly to the next block, DENY makes the escape via the named conditional part.

The scope of the names in a POSIT is statically limited to its body; escapes to irrelevant, external conditional actions are, therefore, impossible. The names of POSITs or their conditional parts may be passed as actual parameters to procedures so that they may PROVE or DENY on behalf of their callers. (The formal parameter would be of type POSIT or DENIAL.)

Does it work?

Figures 2-4 show a simple program. As a flowchart, the problem is easily understood; it's to about listing Products, from a file, and their Suppliers. Suppliers are few enough that they can be looked up in a table. The programmed versions are done in Figure 3 with GO TOs and in Figure 4 with POSITs to show how easy they are to use (if you relax and concentrate).

Let's check how well some of the problems and requirements have been answered:

- POSIT allows escapes from the current block to some later block, up through any number of levels.
- The place to which escape is made - the destination - is chosen according to the cause.
- Branches cannot be made, except to points in, or at the ends of, surrounding blocks.
- There is no need for flags.
- Called procedures can cause an escape; thus hardware-related interrupt handlers can act to change the course of a program.

It looks good so far, the pragmatic requirements of real code have been satisfied, and the dangers of GO TOs avoided. Maybe, when they threw out the GO TO, they were a little too quick to throw out the label.

Two more areas to check are recursion and multi-tasking. Recursion is not really a problem, because the executions of the called procedures (self or other) are synchronous. But suppose that the main body of a POSIT has initiated some asynchronous tasks and - while doing some work of its own - PROVES or DENYS itself. What should happen to the still-working tasks? My answer is that the operating system should terminate them, preferably by a DENY of their outermost POSIT, and they are forgotten about (but I'm prepared to discuss it).

```
...
LOOP
  ReadNext (ProdFile, ProdRec)
  IF ProdFile.FileStatus = EOF THEN
    Writeln ; Writeln ; Writeln ("End of list") ;
    GO TO EndFile
  END ;
  IF ProdFile.FileStatus <> OK THEN
    Writeln ; Writeln ; Writeln ("Product file error: ") ;
    Writeln (ProdFile.FileStatus, 0) ;
    GO TO Windup
  END ;
  Writeln ; Writeln (ProdRec.ProdCode, 7) ;
  Writeln (ProdRec.ProdDesc) ;
  i := LookUp (Supplier, ProdRec.SuppCode) ;
  IF i > 0 THEN
    Writeln (SupplierName [i]) ;
  ELSE
    Writeln ("Supplier name not found for code: ") ;
    Writeln (ProdRec.SuppCode, 0) ;
  END ;
REPEAT ;

EndFile:
  Close (ProdFile) ;
Windup:
  ...
```

Figure 3 - The problem solved with GO TOs

Computer scientists had long explored styles of programming including functional (eg LISP), assertional (eg ancient RPG and modern PROLOG), a massive variety of procedural and special purpose languages and, more recently, neural and vegetable approaches. But in the mid 1960s attention was focused on procedural languages (quite properly so, because most programmers were using them). In this period - and ever since - the academics have searched for purer structures; their constant

bugbear has been the GO TO. In 1966, Böhm and Jacopini introduced an algebra for programming based on the idea of nested blocks of code, in which each block had one entry point and one exit. It used only the structures Sequence, IF and WHILE. These authors admitted that not all programs could be expressed in their algebra. In 1968, Edsger Dijkstra's famous letter 'GO TO Statement Considered Harmful' stirred up much ardour in the hearts of purists and soon it was proved, by others,

that 'every flowchart program can be written without GO TO statements by using WHILE statements'.

High-flown academic debate was all very well, but it took a second school of thought in favour of leap-free code to popularise the notion 'on the ground'. There was a growing band of program provers (such as Floyd in the early '70s), who had a need to limit the damage to their methods that a wayward GO TO could inflict. It was clear that well-structured programs were much easier to understand. The GO TO-less view was promoted and taken up by the makers of increasingly large and complex software systems.

In general, languages continued to provide a GO TO, but there began to appear some varieties which left it out. In these languages, other techniques were introduced to replace it. The most common, in one form or another, was 'BREAK' or 'EXIT'. BREAK caused a transfer of control from within one block to the beginning of the next block at the same level; mostly it was used to provide a premature escape from a loop. One variation which I have seen had

```

POSIT <label> :                               --- labelled block

[<sequence>                                ]-- main body with
PROVE <label> | DENY <label>.<c-label>      ) different kinds
<sequence>] ...                            ) of escape

[DENIAL <c-label> :                          --- any number of
<sequence> ] ...                          conditional parts

END POSIT <label>                           --- labelled end

(Square brackets show optional sections; vertical bar shows alter-
natives; ellipses (...) show sections which may be repeated.)

```

Figure 1 - A POSIT block in outline

Don't buy a Prospero Compiler if you enjoy tracking down obscure bugs in your code.

Prospero takes all the fun out of debugging, by doing a lot of it for you. Prospero compilers can detect an out-of-range variable, an invalid pointer dereference, or an invalid array access as soon as it occurs, so you don't get the chance to spend hours trying to locate such problems yourself. Once you've removed all the bugs, turn off the checks, turn on the global optimizations, and see how fast your code goes!

Prospero Fortran version iid 2.1 for DOS only – £240

Prospero Fortran version iio 2.1 for OS/2 and DOS – £320

Prospero Pascal version iid 5.2 for DOS only – £240

Prospero Pascal version iio 5.2 for OS/2 and DOS – £320

Prospero make a range of standard-conforming Pascal and Fortran-77 compilers for a wide range of computers.

For further information on any of these products, please phone or write to the address below.

Prospero Software

LANGUAGES FOR MICROCOMPUTER PROFESSIONALS
190 CASTELNAU, LONDON SW13 9DH, ENGLAND TEL: 081-741 8531 FAX: 081-748 9344

IN

CASE.

YOU'RE INTERESTED IN QUALITY!

C.A.S.E. – Computer Aided Software Engineering.

VERILOG is setting new standards in real-time software development, with a range of CASE tools that addresses each stage of the development life-cycle, from specification through to testing and validation.

AGE is a powerful, integrated workbench for requirements analysis, specification and real-time system design. Built-in simulation and verification features ensure the quality and integrity of the system.

LOGISCOPE is the only software testing tool which provides complexity and test coverage results for

a full range of development languages. Improved code quality and test effectiveness reduces the cost of software development, testing and maintenance.

VERILOG's software development tools provide powerful features to enhance quality and improve productivity, transforming what was an unstructured and insular process into a formalised industrialised procedure.

VERILOG's tools are already in use on numerous major projects in the UK and Europe. For more details on our complete product range call William Rigg on **081-940 2212**.

OS/2 Dynamic Link Libraries

Dynamic linking is a concept familiar to Windows programmers; but you have to meet them under OS/2 to see them at their best, says Graeme Burton.

The concept of Dynamic Link Libraries (DLLs) is fundamental to OS/2: the operating system kernel itself is packaged as a series of DLLs. However, the documentation supplied with the various toolkits and compilers is hard to assimilate, and spread widely around the manuals. The official texts are also short on practical advice. There are no indications of where problems are likely to occur, or how to solve them.

In this article, I shall explain the concepts behind DLLs, and how to build and debug

programs which incorporate DLLs. I will also discuss some common problems and their solutions. Although I am mainly aiming at C programmers using either the IBM C/2 compiler V1.1 or the Microsoft C compiler V5.1, this does not preclude the use of other languages or compilers - DLLs can be written in any language. However, these are the recommended tools for building OS/2 applications.

Concepts

Under the MS-DOS operating system, relocatability of the various system services is achieved by using software interrupts. Shared code, such as the C runtime libraries, are combined with the application at link time to form part of the .EXE or .COM program file. OS/2 provides a mechanism for applications to resolve references *after* linking has been performed, thus achieving run time relocatability. This is the Dynamic Link Library, or DLL.

DLLs are stored in conventional OS/2 binary files with the extension '.DLL'. When a function contained in a DLL is required, the file is loaded into memory by the loader task, in much the same way as a .EXE file is loaded. Thus, if a program uses a DLL, the executable image is now split into two files: the .EXE file and the DLL file. The functions contained in a DLL are shared between processes by means of the Global and Local Descriptor Tables; the GDT and LDT. This means that only one copy of the code is required.

To see what goes on, let's examine the C function `printf()`. If 15 programs use this function, each one has its own copy of it in the executable file. This is not too serious in terms of disk space requirements. However, when we move from the single tasking MS-DOS environment to the multi-tasking OS/2 environment, and all 15 copies of the program are running at the same time, the `printf()` function will consume 15 times as much RAM,

which is definitely bad news. The cure is to place this function in a DLL. Only one copy of the code is loaded into memory, and all tasks share this copy. Further benefits occur when updating packages and code libraries. If code is packaged in the form of DLLs, upgrades can be supplied to customers by providing a refresh of the DLLs. No refresh of the main executable module is required.

A DLL may be loaded into memory in two ways. The first is to define the procedure required from the DLL when linking your program, either by means of an import library or in a definitions file. The operating system will then load the DLL (if it isn't already in memory) when the .EXE is loaded. The second method is to load the DLL into memory under program control, by using the OS/2 system function `DosLoadModule()`, and locate the entry points required from the DLL with the `DosGetProcAddress()` function. Using this method, a DLL can be removed from memory when it is no longer required (and thus free the RAM it used) by making a call to the `DosFreeModule()` function. OS/2 maintains a usage count of DLLs: when a process links to a DLL, either implicitly or via `DosLoadModule()`, the usage count is incremented. When the link is broken, the usage count is decremented. When the usage count goes to 0, the DLL is unloaded from memory.

The DLL implementation provided by OS/2 forces certain constraints upon users. The internal segment register assumptions made by the compiler in the various memory models (Large, Medium etc) no longer apply. With a non-DLL program, the DS and SS registers are assumed to be the same: with a DLL, this is not the case. This is because the library procedure uses its own data segment, set by the loader when the DLL is brought into memory, and uses the calling procedure's stack segment. As a result, a short prologue has to be included at every routine's entry

```

/* *****
/* This is the main program for the
/* triangle application. It accepts the
/* arguments from the command line.
/* *****
#include <stdlib.h>
#include <stdio.h>

/* Forward-declare
   the triangle function */
int pascal far
   triang ( double, double, double );

int main(argc, argv, envp)
int argc;
char *argv[];
char *envp[];
{ /* Sides of triangle
   read from command line */
   double a,b,c;

   /* Check the arg count is correct
      - quit if it isn't */
   if (argc < 4) {
      printf ( "Usage: TRIANG a b c\n"
              "where a, b, c are the sides"
              "of the triangle\n");
      return ( -1 );
   } /* endif */

   /* Unpack the arguments */
   a = atof(argv[1]);
   b = atof(argv[2]);
   c = atof(argv[3]);

   /* Call the triangle procedure
      and print the results */
   switch (triang( a, b, c )) {
   case 0:
      printf ( "Equilateral triangle\n");
      break;
   case 1:
      printf ( "Isosceles triangle\n");
      break;
   case 2:
      printf ( "Scalene triangle\n");
      break;
   default:
      printf ( "Impossible triangle\n");
      break;
   } /* endswitch */

   /* Return to OS/2 */
}

```

Figure 1 - EXE source for the TRIANGLE program

SATISFY YOUR APPETITE FOR KNOWLEDGE WITH 15 SUBSTANTIAL COURSES.



Intel Customer Training can now offer you even more to digest.

Some succulent new courses to add to our highly successful Microprocessor training.

People who are hungry for more information can now sit down and enjoy the latest training in PC Networking or UNIX.

The first hundred to attend can wash it down with some free champagne.

The venue is our training facility at our headquarters in Swindon.

With over £1 million worth of equipment and some of the most experienced training staff around, everything should go down rather well.

Or if you like, we can provide a take-away service by training you at a venue of your choice.

Call Dee Blanche-Horgan, our Training Administrator on (0793) 696539 or return the coupon for the complete Intel Training Brochure. You'll find it's a tasty little starter.

Please send me more information about the new Intel Training Courses.

NAME

POSITION

COMPANY

ADDRESS

POSTCODE

TELEPHONE NO.

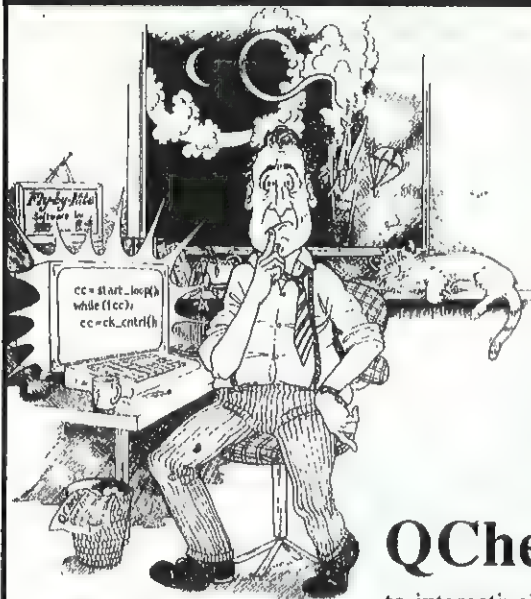
Return to: Intel
Customer Training,
Pipers Way, Swindon
SN3 1RJ.

EXL 7/90

intel

CIRCLE NO. 080

THE SOFTWARE TESTING NIGHTMARE ?



- Are you serious about the way you develop programs?
(Of course I am)
- Do you test your programs?
(Doesn't everybody?)
- How much do you test your programs?
(I try out all, well most, of its functions)
- Do you look forward to testing your programs?
(Who does?)
- Can you quantify how much you test your programs?
(How could I, even if I wanted to?)
- Have you considered the implications of not being able to demonstrate that you have fully tested your program?
(...Ah, um...not really yet)

QCheck is a unique program that enables professional software developers to interactively find out how much their program has been exercised. It also provides objective evidence in the form of listings showing unexecuted lines. For DOS high level language compilers. A single user licence is £150.00+VAT (how much does just one undetected bug cost you?)

Also - **ETP** an Epson- to Postscript- translation utility that allows existing applications to print to your Postscript laser printer, whether connected locally or on your network. A single user licence costs £55.00 + VAT.

Telephone 0883-341697, Fax 0883-341343

Seltek Ltd., Seltek House, 38 Westway, CATERHAM, Surrey. CR3 5TP.

SELTEK

CIRCLE NO. 081

DOS PC-NFS UNIX

If you can't beat it - join it

Sun PC-NFS - the ultimate UNIX/DOS network integration tool, using standard Ethernet, TCP/IP and NFS protocols.

- mount DOS file systems on Unix servers
- use DOS word processors to edit Unix files
- access remote printers across a network
- log into remote systems
- send and receive email across the world

PC-NFS

manufactured in the UK under licence from Sun Microsystems by Unipalm
Europe's foremost supplier of TCP/IP X Window and NFS solutions
Also available for **PC-NFS** -
Tell-PC network phone

Unipalm Limited 145-147 St Neots Road • Hardwick • Cambridge CB3 7QJ • England
• Tel 0954 211797 • Fax 0954 211244 • Intl. Tel +44 954 211797 • Intl. Fax +44 954 211244

PC-NFS and NFS are registered trademarks of Sun Microsystems, Inc. UNIX is a trademark of AT&T

Unipalm

CIRCLE NO. 082

Figure 2 - DLL source for the TRIANGLE program

```

/*****
/* This function calculates what sort
/* of a triangle has been. It returns
/* 0 for an equilateral triangle,
/* 1 for an isosceles triangle,
/* 2 for a scalene triangle,
/* 3 for an impossible triangle.
*****/
int pascal far triang ( a, b, c )
/* Input sides of the triangle */
double a,b,c;
{
    /* Check for impossible triangle */
    if ((a <= (b + c)) &&
        (b <= (a + c)) &&
        c <= (b + a)) {

    /* Check for equilateral triangle */
    if ((a == b) && (a == c)) {
        return ( 0 );
    } else {

    /* Check for isosceles triangle */
    if ((a == b) ||
        (a == c) ||
        (b == c)) {
        return ( 1 );
    } else {
        return ( 2 );
    } /* endif */
    } /* endif */
    } else
        return ( 3 );
}

```

point to save the current DS and replace it with the DLL's. The calling procedure's DS is restored at function exit by a similar mechanism. There are several ways to force the inclusion of this code. The recommended method is to compile with the /Alfu memory model. An alternative approach, which allows function by function control at source code level, is to declare functions with the `_LOADDS (= LOAD DS)` keyword.

When a DLL is created, you have choices to make about both initialisation and data sharing. A user supplied initialisation procedure can be called when the DLL is loaded, although this requires an assembler stub. Instructions supplied through a definitions file at the link step can force this procedure to be called when the DLL is first loaded into memory, or each time a new module links to the DLL. The data used by the DLL can be global, ie shared between all processes that are using the DLL, or local, in which case each instance has a different data segment. Again, instructions in the definitions file control the data sharing.

As well as functions, data can be stored in a DLL. A variable to be exported must be specified in the definitions file, with its name preceded by `'_'`. DLLs can also be used to contain PM resources: if the resource is stored in a DLL, the module handle of the DLL is passed to the PM calls `WinLoadDlg()`, `WinLoadAccelTable()`, `WinLoadMenu()`, `WinLoadMessage()`, `WinLoadPointer()` and `WinLoadString()`. The module

handle for a DLL can be obtained by calling either of the OS/2 functions `DosLoadModule()` or `DosGetModuleHandle()`.

Library Functions

The DLL developer cannot use the standard C run-time library functions. These were not written for use with DLLs; any attempt to link them to a DLL produces an unresolved external main. It is possible to use the standard run time libraries by kludging assembler links, but it is far easier and more sensible to use the libraries specially provided for this purpose: the `LLIBCDLL` and `CRTLIB` libraries.

The `LLIBCDLL` library is linked statically with the DLL, and contains all the C functions provided by the standard C library. It is intended to be used with the large memory model. The functions within this library are not re-entrant (for example, the `fevt()` function, which converts a floating-point number to a character string, uses a static buffer), and so multi-threaded code linked with this library will not work. You should note that DLLs which use `LLIBCDLL` cannot be freed by calling `DosFreeModule()`. This is because the start-up library code installs an entry in the process's exit list. Once these DLLs are loaded, they stay loaded until the invoking process is terminated.

DLLs which are linked with the `LLIBCDLL` library must be compiled with the /FPa switch, specifying the alternate math(s) library, if they use floating point operations, otherwise the C library floating point functions will not work. This is because the called function (in the DLL) does not know what state the calling application has left the 80x87 processor in - and so cannot use it.

The second library is the `CRTLIB` (C Run-Time Library), which consists of a package from all the C libraries, bundled into a DLL. This library is re-entrant, and allows floating point processor instructions to be used. The `CRTLIB` DLL can be linked to produce either .EXE or DLL modules. To create a DLL module, you simply define the symbol `DLL` to the compiler (using the `#define` command in the source code). You must

include the special multi-threaded versions of the header files when you are going to use the `CRTLIB` library (generally stored in the `INCLUDE\MT` subdirectory). This is because the C library function declarations are slightly different for the re-entrant libraries.

You must also specify the `NOIGNORECASE` switch (/NOI) when linking. The linker converts function names to upper case by default, but `CRTLIB.DLL` is built with lower case function names. If you don't use the /NOI switch, the C library functions are not found when you run your application. Finally, you must remember to include the file `CRTDLL.OBJ` when linking the DLL, and `CRTEXE.OBJ` when linking the .EXE file. You can ignore `CRTEXE.OBJ` if you only use functions which require no initialisation, such as the string manipulation functions. See Figure 5 for the link information required.

The `CRTLIB` DLL is created under user-control, by a command file supplied with the compiler (`CDLLOBS.CMD`). This lets you include only those functions which your applications need - to save disk space and RAM - although I would caution against it. It is better practice to place all the C functions into the `CRTLIB` DLL, otherwise you will get into trouble when the requirements of your application and someone else's, installed on the same machine, do not coincide. The file containing this DLL is not shipped with the operating system, so if your application uses the `CRTLIB`, ensure that a copy of it goes out with your software.

Building DLLs

The source code for DLLs is written in the same way as for .EXE modules. The main difference lies in the definitions (.DEF) file supplied to the linker. The definitions file contains ASCII text which identifies the procedures to be exported from the DLL being linked. A definitions file can be supplied for programs as well as DLLs: Figures 1 to 3 show the .EXE triangle program, with definitions files for both the application (.EXE) and the DLL.

```

whole: triang.dll main.exe

triang.dll:  triang.obj
            link /co /nod triang.obj,triang.dll,,doscalls llibcdll,triang.def;

triang.obj:  triang.c
            cl /Zi /Od /c /W3 /Alfu /G2s /FPa triang.c

main.exe:   main.obj
            link /co /nod main.obj,main.exe,,doscalls llibce,main.def;

main.obj:   main.c
            cl /Zi /Od /c /W3 /AL /G2s /FPa main.c

```

Figure 3 - MAKE file for the triangle application

Figure 4 - Modifications to triangle EXE for DosLoadModule

```

/*****
/* This is the main program for */
/* the triangle application. */
/*****
#include <stdlib.h>
#include <stdio.h>
#define INCL_DOSMODULEMGR
#include <os2.h>

int main(argc, argv, envp)
int argc;
char *argv[];
char *envp[];
/* Sides of triangle */
double a,b,c;
/* Buffer for
LOAD MODULE errors */
CHAR FailBuff(255);
/* DLL module handle */
HMODULE ModHandle;
/* Pointer to TRIANG function */
int (pascal far * triang)
(double, double, double);

/* Check the arg count is correct
- quit if it isn't */
if (argc < 4) {
printf ( "Usage: TRIANG a b c\n"
"where a, b, c are"
"sides of the triangle\n");
return ( -1 );
} /* endif */

/* Unpack the arguments */
a = atof(argv[1]);
b = atof(argv[2]);
c = atof(argv[3]);

/* Do the dynamic link
of the triangle DLL */
DosLoadModule ( FailBuff,
sizeof(FailBuff),
"TRIANG1", &ModHandle );
DosGetProcAddr ( ModHandle,
"TRIANG", &triang );

/* Make an INDIRECT call to the
triangle procedure */
switch (triang( a, b, c )) {
case 0:
printf("Equilateral triangle\n");
break;
case 1:
printf("Isosceles triangle\n");
break;
case 2:
printf("Scalene triangle\n");
break;
default:
printf("Impossible triangle\n");
break;
} /* endswitch */

/* Free the triangle DLL off */
DosFreeModule( ModHandle );

/* Return to OS/2 */
return ( 0 );
}

```

The important definitions file directives for DLLs are the `LIBRARY`, `EXPORTS` and `DATA` commands. The `LIBRARY` statement identifies the file as a DLL, so that the linker knows not to search for a main entry point. The statement can optionally contain the `INITINSTANCE` or `INITGLOBAL` modifiers. These determine whether the library initialisation procedure is called once only, or each time a new process links to the library. The `EXPORTS` directive informs the linker which function addresses are available as entry points. Finally, the `DATA` statement indicates whether data is shared by all processes using the DLL, or private to each process. This is controlled by the `SHARED/NONSHARED` and `SINGLE/MULTIPLE` parameters. Here is the definitions file for the example .EXE

module:

```

IMPORTS
TRIANG, TRIANG

```

and here is the file for the DLL:

```

LIBRARY
EXPORTS
TRIANG

```

Figures 1 to 3 contain the source code and MAKE files to build .DLL and .EXE files which solve the famous .EXE triangle program, as featured in *The Third Side*. Figures 4 and 5 show the modifications required to use CRTLIB, and to load the DLL with the `DosLoadModule()` and `DosGetProcAddr()` system calls.

Debugging DLLs

Dynamic Link libraries can be source-level debugged with the CodeView debugger, provided that they are not accessed via calls to `DosLoadModule()` and `DosGetProcAddr()` - in this case the physical addresses that CodeView requires are not available until the module has been loaded into memory. It is possible to get round this by `IMPORTING` an arbitrary address from the DLL being debugged. This forces the DLL to be loaded along with the executable module, and addresses are resolved correctly. Once the debugging is complete, the import can be deleted from the definitions file.

When debugging DLLs, the `/L` (Library) switch is used to identify the DLL to CodeView. Multiple libraries may be debugged by specifying each one on the command line. The `/L` parameter must be specified on the command line before the executable file being debugged. If you put it afterwards, CodeView assumes that it is a parameter of the program being debugged. There must be a copy of the DLL in the current directory for CodeView to enter the library successfully.

Common Problems

So, you've built your EXE and DLL without any compile or link errors. You now try to run it. The system prints the message `Unable to find the file .` and returns you to the command prompt. What do you

do next? This is typical of the often cryptic, and sometimes incomprehensible, error messages which OS/2 produces when DLL problems occur. I have spent many hours trying to track down such problems. Here are a few of the best.

The system cannot find the file ., or alternatively, 'The system cannot find the file XYZ.DLL' is caused by the DLL not appearing in the `LIBPATH`. When the loader loads a DLL into memory, it searches the directories specified by the `LIBPATH` statement in the `CONFIG.SYS` file. If the DLL is not in a subdirectory specified in your `LIBPATH`, you will receive the above message. Either move the DLL into a directory in the `LIBPATH`, or to update the `LIBPATH` and reboot.

The message 'SYS2070: Unable to demand load the application's segment' is my favourite for incomprehensibility. This is generally caused by a missing entry point. It often occurs as a result of switching to the `CRTLIB` library, and forgetting to link with the `/NOI` option. The solution is to ensure that all the entry points you require are actually stored in the DLL, and the case of identifiers matches. The 'SYS2070' error message can also occur when a DLL has no static data segment. This can be cured by defining a single static variable in one of the source files.

Problems can occur at the link stage. If you have problems with unresolved externals or multiply-defined symbols, ensure that you are linking with the correct libraries: specify `/NOD` to ensure that the default C libraries aren't being searched, and include either `LLIBCDLL` or `CRTLIB` as the libraries. Don't include any of the standard C libraries, such as `LLIBCA`, `LLIBC7` etc.

One final point to watch for is the unresolved symbol `__FAC`. This occurs when returning floating point numbers from functions. The solution here is either to pass a pointer to the floating point variable to the function, or to declare the function of type `PASCAL`.

EXE

Graeme Burton is a freelance programmer, currently working for IBM.

```

whole: triang.dll main.exe

triang.dll: triang.obj
link /co /nod /noi triang.obj d:\ibmc2\lib\crt.dll.obj,triang.dll,,doscalls
crtlib,triang.def;

triang.obj: triang.c
cl /Zi /Od /c /W3 /AL /G2s /Id:\ibmc2\include\mt /DDLL triang.c

main.exe: main.obj
link /co /nod main.obj,main.exe,,doscalls llibc;

main.obj: main.c
cl /Zi /Od /c /W3 /AL /G2s main.c

```

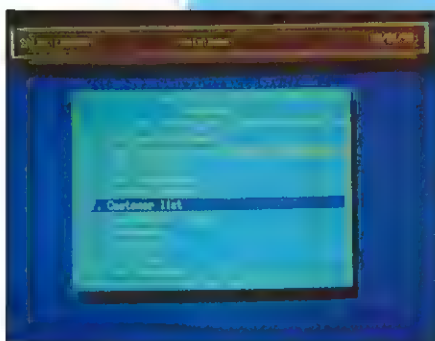
Figure 5 - MAKE file for DosLoadModule and CRTLIB

dBASE USERS!

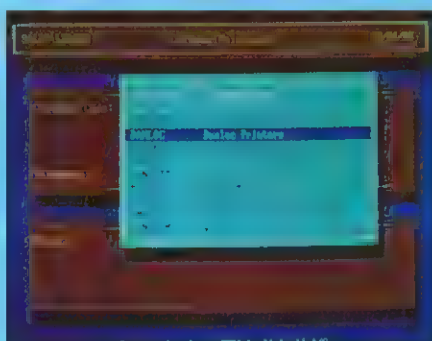
**How many days would it take to program, test and debug this system?
With Sycero and Clipper you could do it in HOURS!**

Clipper is the applications development standard for PC database management systems. Clipper's open architecture and superior networking capabilities let you create more sophisticated and powerful applications.

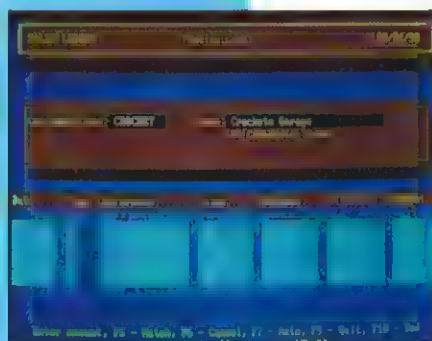
Sycero is a complete applications generator for Clipper. Making use of all of Clipper's advanced facilities, Sycero lets you develop PC applications in a fraction of the time it would take to do manually.



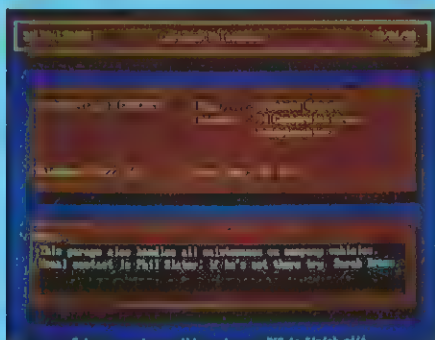
Pulldown, bouncebar and Novell style menus produced in minutes.



Overlay a browse window with a function key in just five lines of code.



Automated transaction handling.



Memo fields give you word processing in a window on your screens.



Complete & powerful report generator for forms, lists and more complex invoices/statements.

Sycero with Clipper - the perfect development environment.

Please send me further details about Sycero & Clipper:

Name:

Address:

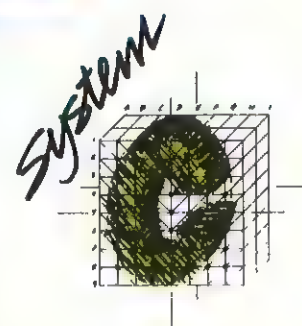
.....

.....

Tel: Fax:

System C Ltd., 60-61 High Street, Maidstone, Kent ME14 1SR
Tel:0622 691616 Fax:0622 691241

EKE 7/90



CIRCLE NO. 114

dBASE® IS A REGISTERED TRADE MARK OF ASHTON-TATE.

Turbo Pascal OOP Text Windows

We have long been hearing about the advantages of class libraries. There haven't been many practical examples about, though - so David Bolton wrote his own.

Turbo Pascal V3.0 and later support a simple form of text windowing on 80 column CGA/Mono screens. Borland's CRT unit allows a section of screen to be 'cordoned off'. Many of the CRT procedures and functions work relative to this window, so that cursor positioning, scrolling, colours and text I/O are all confined within it. It is crude, but it handles much of the donkey work and simplifies the task of adding a decent front end.

A couple of years ago I wrote a simple Windows system, based around the Move procedure (which copies contiguous blocks of memory). It was handy for displaying error messages and help screens, but it was not a full windowing system.

When Turbo Pascal V5.5 appeared, I decided to redesign my existing windows code, making it object oriented and also try to improve its performance. The original code used an array of window records, and windows could only be opened and closed in order. I also wanted my OOP code to be flexible, so that when I could afford a better micro, the code would take advantage of features like VGA without having to start all over again. With such thoughts in my head, I took another look through my non-OOP code, and did the only thing possible: junked it.

Design Thoughts

What is a window? Just an area of screen defined by co-ordinates, size, and a title. Add in two pointers to link into other windows and that is the simplest class of window, defined in my program as the object `ClassWindow`. Co-ordinates and dimensions were made word sized so that the objects could be used with graphics screen modes.

The design handles multiple windows and lets them be opened and closed in any order. Windows can be moved about, resized and any window moved from behind any other. The text in the window isn't preserved when windows are moved about, though with a little extra work this could be added.

The structure I chose was a linked list of window objects, with each object dynamically created. I had originally thought of each window as a statically declared object, accessed with code along the lines of `Window1.Init`. However, as I had decided to store the objects as a linked list, purely static windows were out of the question. Accessing an object had to be done either indirectly, as in `Window1^.Init`, or through a pointer (internal to the object) to the Window data.

The second method was rejected, as it would require the object type defined outside the object definition. This would have lost much of the advantages of OOP, such as inheritance. I toyed with defining the data part as an object, but that line of thinking brought me back to my first idea. This was the indirect method, with objects created on the heap. Borland's OOP implementation supports this with extensions to `New` and `Dispose`.

The list is a double chain with two pointers in each object. The lists firmly anchored between a head and tail pointer. When a Window is opened, it is added to the end of the list, using the tail pointer. This allows easy access to any object from either direction. It so happens that the head pointer isn't used in my program, but I've left it in.

I tried to isolate those elements which are general to any windows (graphic or text based) and put those in `ClassWindow`. `OpenWindow` hooks a window onto the end of the list, and `CloseWindow` removes it. These apply to any window system derived from `ClassWindow`.

`ClassWindow` and its methods do not create a fully working window system; they create an extensible structure which can be built on. Attributes that must be added in descendant objects include the method of storing the screen under the window, colours and border type. These are handled in different ways in a text or graphic system. I have left out other attributes, like shadowing and growing a window, to keep the code compact.

The object type `ColTextWind` was derived from `ClassWindow` to manipulate windows on a text screen, with colours, borders and text saved out from beneath the window. Though designed for colour systems, with very slight modifications it will run on mono systems. (In fact, it does in its present form, but is

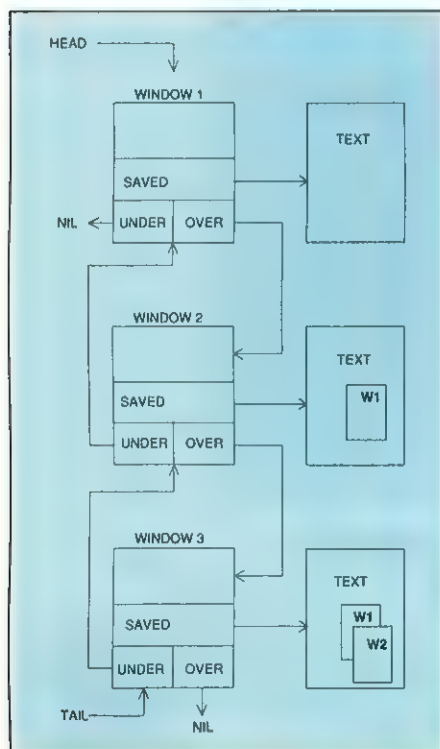


Figure 1 - Structure diagram with three open windows

TopSpeed[®] C, Modula-2 and Pascal



Finally, a professional set of compilers designed to work together. All TopSpeed compilers share the same development environment and use the same optimizing code generator and run-time system.

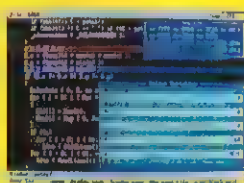
	TopSpeed [®] C Version 1.00	Microsoft [®] C Version 5.1	Turbo C [®] Version 2.0	Watcom [®] C Version 7.0
100% ANSI compatible*	✓			
Integrated environment	✓		✓	
Pass parameters in registers	✓			✓
Expand any function as inline code	✓			
Supports OS/2	✓	✓		
DOS Dynamic Link Libraries (user's code linked at load time)	✓			
Smart linking (only referenced code and data linked into .EXE)	✓			
Type-safe linking (function parameters and memory model checked at link time)	✓			
Fully automatic make works across libraries	✓			
Time-sliced scheduler for multi-tasking under DOS	✓			
Short pointers in any segment	✓			
Hypertext help with library online	✓		✓	

*Written by Neil Martin of the British Standards Institution (BSI) and printed in Personal Computer World June 1985, page 241.

etc., code searches, integrated
text and saved environment.

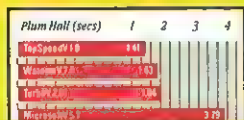


• Cross-platform debugger
• Windows support
• Assembly debugger



In the U.S.A. contact:
Jensen & Partners Int.
1101 San Antonio Road
Suite 301
Mountain View
CA 94043
USA

Standard Edition \$199
Extended Edition \$399
OS/2 \$495



Standard Edition—

Optimizing DOS compiler, integrated source-level debugger, smart linker, automatic make, complete standard library, 6 memory models, BGI interface, time-sliced scheduler, MS-DOS, BIOS interface, and more.

Extended Edition—

Standard Edition plus full source code to libraries, post-mortem debugging, MS Windows support, support for DOS DLLs (overlays), assembler, disassembler, profiler, DOS call monitor and more.

OS/2 Edition—

Extended Edition (except for DOS compiler) plus ability to generate DOS executables, full support of OS/2 kernel & PM calls, OS/2 smart linker, fully automatic generation of DLLs, source for protected mode libraries, and more.

Benchmark measured by Mark
Henderson, November 1985.
Tested on IBM PC compatible hardware.
Worldwide Communications Ltd.

Each compiler is available in three editions:

Standard Edition £149

(DOS Compiler & VIO)

Extended Edition £295

OS/2 Edition £370

64-page Technical Specifications
booklet describing TopSpeed
compilers is available
upon request.

**ALL PRICES
EXCLUDING VAT**

To Order:

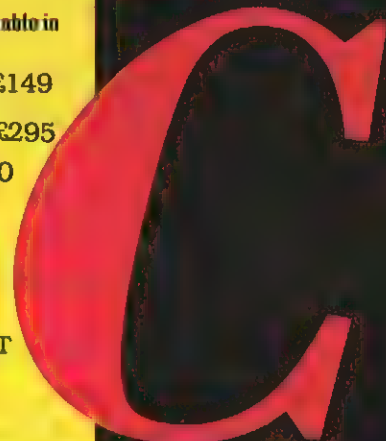
TEL: 071-253-4333

FAX: 071-251-1442

Call on shipping & handling
charges & volume discounts.
VISA/MC accepted.

**30-day unconditional
money-back guarantee.**

TopSpeed



Jensen &
Partners
UK Ltd.

Unit 21
63, Clerkenwell Road
London EC1M 5NP

TopSpeed and TechKit are trademarks of Jensen & Partners International. Other brand and product names are trademarks or registered trademarks of their respective holders.

CIRCLE NO. 115

PURE MAGIC★

COMPUTER AIDED PROGRAMMING
APPLICATION GENERATOR &
RELATIONAL DATA BASE

BEYOND 4GL!

— THE GOALS —

- ★ Well-engineered applications for end-users
- ★ Up to 90% reduction in development effort
- ★ Minimal on-going maintenance costs

— THE SOLUTION —

MAGIC

Computer Aided Programming
Application Generator and Relational Database

— THE MEANS —

- ★ Fully integrated Database and Application ENGINE
- ★ For developers, the ENGINE drives a special application - The Computer Aided Programming Application - eliminating the need for any 4GL.
- ★ For end-users, the ENGINE drives the developed application with user-friendly facilities conveniently specified by the developer through easy-to-use tables

— THE PROOF —

MARCH '89, AUSTRALIA: **MAGIC** - Outright winner in
PC WEEK Database Challenge

JANUARY '89, USA: **MAGIC** - Shared Top Honours in
PC WEEK Poll of Corporate Satisfaction

"There is no competitive product"

(PC TECH JOURNAL)

"... this program is a stellar performer"

(PC WORLD)

"One of the most powerful DBMS
packages available" (PC MAGAZINE)

— THE ULTIMATE PROOF —

Thousands of experienced, knowledgeable,
professional developers in over 20 countries have
switched from better-known 'brand-name' products
to **MAGIC**

— THE ENVIRONMENTS —

OPERATING SYSTEM: PC-DOS/MS-DOS 2.0 and up

HARDWARE: IBM-PC/XT/AT, IBM PS/2 and full
compatibles, 640K RAM

NETWORK SUPPORT: Any DOS 3.1 compatible
network including Novell Netware and IBM
Token-Ring

Versions also available for Unisys-BTOS,
Convergent-CTOS.



UK ENQUIRIES:

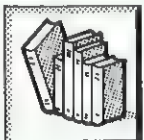
MSE (UK) LTD, Fulton House, Fulton Road,
Empire Way, Wembley, Middx HA9 0TF
TEL: 01-902 8998 FAX: 01-903 1465

All trademarks and copyrights are acknowledged

CIRCLE NO. 111

X Window System
Release 4 of X11 is now available

• IXI Limited will help you exploit this
important software standard . . .



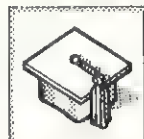
BOOKS & MANUALS

Specialist books for Software Developers
The new O'Reilly Toolkit Tutorials, plus User's Guide
to X, Programming Tutorials for Xlib and more.



SOURCE CODE

Release 4 of the MIT source code with X servers for all
major UNIX platforms. Available on 0.25" cartridge
tape and IBM PC-AT 5.25" floppies (UNIX tar files)



TRAINING

Our well established training courses cover:
1 day Strategic Overview of X
4 days Programmer's Workshop with hands-on training.



0223 462131
+44 223 462131

IXI



IXI Limited · 62-74 Burrell Street · Cambridge CB1 10J · England · TEL: +44 223 462131 · FAX: +44 223 462132

*X Window System is a trademark of MIT

CIRCLE NO. 083

Figure 2 - Window objects code

```

(SR+,S+,I+,D+,F-,V-,B-,N- )
($M 10000,0,60000 )
PROGRAM EXMIND;

USES CRT,DOS;

CONST
  BufferSize=4000;

TYPE
  WindowPtr = ^ ClassWindow;
  TitleType = STRING[20];

ClassWindow= OBJECT
  Xpos,Ypos :WORD;
  Xsize,Ysize :WORD;
  Open :BOOLEAN;
  Title :TitleType;
  Over,Under :WindowPtr;
  CONSTRUCTOR Init(NewTitle:TITLETYPE);
  PROCEDURE OpenWindow(Atx,Aty,Xlength,Ylength:WORD);
  PROCEDURE CloseWindow; VIRTUAL;
  FUNCTION IsWindowOpen: BOOLEAN;
  PROCEDURE MoveWindow(TOX,TOY:WORD);
  PROCEDURE CurrentWindow; VIRTUAL;
  DESTRUCTOR Done; VIRTUAL;
END; { ClassWindow }

BorderType= (NoBorder, SingleBar, DoubleBar);
SavedText = ARRAY[1..BufferSize] OF CHAR;
SavedPtr = ^SavedText;
ColType = 0..15;

ColWindPtr = ^ColTextWind;

ColTextWind= OBJECT(ClassWindow)
  Edge : BorderType;
  Saved : SavedPtr;
  WindowCol:BYTE;

  CONSTRUCTOR Init(NewTitle:TitleType;
    TheEdge: BorderType;
    TextCol,Backcol: ColType);
  PROCEDURE OpenWindow(Atx,Aty,Xlength,Ylength:WORD); VIRTUAL;
  PROCEDURE CloseWindow; VIRTUAL;
  PROCEDURE OpenWide; VIRTUAL;
  PROCEDURE RestoreScreen; VIRTUAL;
  PROCEDURE ClearWindow(Textcol,Backcol:ColType);
  DESTRUCTOR Done; VIRTUAL;
END; { ColTextWind }

VAR Head,Tail:WindowPtr;
OS:STRING[80];

CONSTRUCTOR ClassWindow.Init
  (NewTitle:TitleType);
BEGIN
  Title:=NewTitle;
  Over:=Nil;
  Under:=Nil;
  Open:=False;
END;

FUNCTION ClassWindow.IsWindowOpen:BOOLEAN;
BEGIN
  IsWindowOpen:=Open;
END;

PROCEDURE ClassWindow.OpenWindow(Atx,Aty,
  Xlength,Ylength:WORD);

```

```

BEGIN
  Xpos:=Atx;
  Ypos:=Aty;
  Xsize:=Xlength;
  Ysize:=Ylength;
  Open:=True;
  Over:=Nil;
  Under:=Tail;
  Tail:=@Self;
  IF Under = NIL
  THEN Head:=Tail
  ELSE Under^.Over:=Tail;
END;

PROCEDURE ClassWindow.CloseWindow;
VAR Temp:WindowPtr;
BEGIN
  Open:=False;
  IF Under <> NIL THEN
  BEGIN
    Temp:=Under;
    Temp^.Over:=Over;
  END;
  IF Tail=@Self THEN
  BEGIN
    Tail:=Under;
    Head:=Tail;
  END
  ELSE
  IF Head=@Self THEN
    Head:=Over;
  END;
END;

PROCEDURE ClassWindow.MoveWindow(TOX,TOY:WORD);
BEGIN
  CloseWindow;
  OpenWindow(TOX,TOY,Xsize,Ysize);
END;

PROCEDURE ClassWindow.CurrentWindow;
BEGIN
  CloseWindow;
  OpenWindow(Xpos,Ypos,Xsize,Ysize);
END;

DESTRUCTOR ClassWindow.Done;
BEGIN
  IF Open THEN CloseWindow;
END;

VAR Dummy,ScreenAdd:SavedPtr;
Scoeff:WORD;

($F+FUNCTION HeapFunc(Size:Word):INTEGER;
BEGIN
  Heapfunc:=1;
END;
($F-)

PROCEDURE CopyDummyToScreen;
BEGIN
  MOVE(Dummy^,ScreenAdd^,4000);
END;

PROCEDURE CopyScreenToDummy;
BEGIN
  MOVE(ScreenAdd^,Dummy^,4000);
END;

```

```

PROCEDURE CopyToDummy(S:SavedPtr);
BEGIN
  MOVE(S^,Dummy^,4000);
END;

PROCEDURE CopyFromDummy(S:SavedPtr);
BEGIN
  MOVE(Dummy^,S^,4000);
END;

PROCEDURE Cursor(TopLine, BottomLine : BYTE);
VAR Regs:REGISTERS;
BEGIN
  WITH Regs DO
  BEGIN
    Ax := 1 shl 8;
    Cx := TopLine shl 8 + BottomLine;
    INTR($10,Regs);
  END;
END;

PROCEDURE CursorOff;
BEGIN
  Cursor(48,0);
END;

PROCEDURE CursorOn;
BEGIN
  IF Scoeff=$B000
  THEN Cursor(13,14)
  ELSE Cursor(5,6);
END;

CONSTRUCTOR ColTextWind.Init(
  NewTitle:TitleType;
  TheEdge: BorderType;
  TextCol,Backcol: ColType);
BEGIN
  NEW(Saved);
  IF Saved=NIL THEN
  BEGIN
    ColTextWind.Done;
    Fail;
  END;
  ClassWindow.Init(NewTitle);
  Edge:=TheEdge;
  WindowCol:=(BackCol shl 4)+Textcol;
END;

PROCEDURE ColTextWind.OpenWide;
TYPE CBL=ARRAY[1..6] OF CHAR;
ST80=STRING[80];
CONST
  CHD:CBL=($201,$187,$200,$188,$205,$186);
  CHS:CBL=($218,$191,$192,$217,$196,$179);
  CIBL:CBL=(' ',' ',' ',' ',' ',' ');
  VAR DWR:WORD;

  PROCEDURE DGoToXY(X,Y:WORD);
  BEGIN
    DWR:=(Y+YPOS-1)*160+((X+XPOS-1)SHL 1);
  END;

  PROCEDURE DWrite(S:ST80);
  VAR I:INTEGER;
  BEGIN
    FOR I:= 1 TO LENGTH(S) DO
    BEGIN
      MEM[SEG(DUMMY^):DWR]:=ORD(S[I]);
      MEM[SEG(DUMMY^):DWR+1]:=WindowCol;
      INC(DWR,2);
    END;
  END;

```

limited in its colours.) The procedure `IsCorM` detects where the screen is located, so the routines work on both mono or colour systems. Figure 1 shows the structure with three windows open.

How it works

A 4000 byte shadow screen (called `Dummy`) is first created on the heap. All screen updates are performed in this screen, then copied back to the real screen using the `Move` procedure. This avoids any flicker and makes updates look smooth and professional. In fact, the CGA adapter supports four screen buffers in text mode, which can be used for this purpose. I considered this approach, but adopted the heap method because it is portable to mono screen systems.

Each window object is created by `ColTextWind.Init`, a procedure called from `New`. Its first action is to try and grab 4000 bytes from the heap. If this fails, `New` returns a `Nil` value and `Init` exits through `Fail`,

a Borland extension which handles objects that initialise unsuccessfully. `Fail` deallocates the object and exits. The heap error function `HeapFunc` stops `New` from bombing out if there is not enough RAM, returning a `Nil` value instead.

When a window is opened, `ClassWindow.Open` puts the object on the end of the list. The routine `OpenWide` does most of the work creating the window. Before it starts, it makes its own copy of the shadow screen - which is used as a record of the image before a window is put down.

The window's border is created in the shadow screen and the window cleared before the shadow screen is copied onto the screen. Finally the CRT's unit's `Window` defines the window limits so all screen writes, scrolls are kept within the required region.

The shadow screen routines (`DGoto`, `DWrite` and `DClear`) are equivalent to `GoToXY`, `Write` and `ClrScr` in CRT. I

have written them in Pascal, though they would be best rewritten in assembler for speed. They are just about fast enough on my 8 MHz Amstrad PC1512.

The `Copy...` procedures use Borland's `Move` procedure to transfer 4000 byte blocks between the heap, shadow screen and the screen. These routines could also be rewritten in assembler and the screen underneath the window saved out, rather than the entire screen. However, I was being lazy: I used `Move` as a quick and easy way to save out an entire screen.

When a window is closed, it puts back its copy of the shadow screen using `RestoreScreen`. This takes it, and all open windows above it, off the screen. `ClassWindow.CloseWindow` removes the window from the list, then all windows above it are reopened. This all takes place in the shadow screen which is finally copied back to the main screen. Moving a window involves closing it, changing its co-ordinates and then reopening it. As it has added itself

Figure 2 - Window objects code

<pre> END; END; PROCEDURE DCLear(LX,LY:WORD); VAR S:ST80; I,BYTE; J:WORD; BEGIN S:=COPY(OS,1,LX); FOR I:= 1 TO LY DO BEGIN J:=DMR; DWrite(S); DMR:=J+160; END; END; VAR Lp,IX,IY,Lt,X2,Y2,X3,Y3,X4,Y4:WORD; XS:STRING(40); Chb:Chb; BEGIN (OPENWIDE) CopyFromDummy(Saved); Open:=TRUE; X2:=Xpos+Xsize-1; Y2:=Ypos+Ysize-1; CASE Edge OF NoBorder:Chb:=Chb1; SingleBar:Chb:=Chd; DoubleBar:Chb:=Chd; END; DGoToXY(1,1); DWrite(CHB[1]); Y3:=1; X3:=1; X4:=Xsize; Y4:=Ysize; XS:=''; FOR IX:= 1 TO Xsize-2 DO XS:=XS+CHB[5]; DGoToXY(X3+1,Y4); DWrite(XS); IF Title<>' THEN BEGIN Lt:=LENGTH(Title); IF Lt>Xsize THEN Title:=COPY(Title,1,Lt-2); Lp:=(Xsize-Lt) DIV 2; DELETE(Xs,Lp,Lt); INSERT(Title,Xs,Lp); END; DGoToXY(X3+1,Y3); DWrite(XS); FOR IY:= 2 TO Ysize-1 DO BEGIN DGoToXY(X3,IY);DWrite(CHB[6]); DGoToXY(X4,IY);DWrite(CHB[6]); END; DGoToXY(X4,Y3);DWrite(CHB[2]); DGoToXY(X3,Y4);DWrite(CHB[3]); DGoToXY(X4,Y4);DWrite(CHB[4]); DGoToXY(2,2);DCLear(XSIZE-2,YSIZE-2); </pre>	<pre> WINDOW(XPOS+2,YPOS+2,X2,Y2); END; PROCEDURE ColTextWind.OpenWindow(Atx,Aty, Xlength,Ylength:WORD); BEGIN CursorOff; ClassWindow.OpenWindow(Atx,Aty,Xlength,Ylength); OpenWide; CopyDummyToScreen; CursorOn; WINDOW(Atx+2,Aty+2,Atx+Xlength-1,Aty+Ylength-1); END; PROCEDURE ColTextWind.RestoreScreen; BEGIN CopyToDummy(Saved); END; PROCEDURE ColTextWind.CloseWindow; VAR Temp:ColWindPtr; Otemp:WindowPtr; Absolute Temp; BEGIN CursorOff; RestoreScreen; ClassWindow.CloseWindow; Temp:=@Self; WHILE Temp^.Over<>NIL DO BEGIN Otemp:=Temp^.Over; Temp^.OpenWide; END; CursorOn; END; PROCEDURE ColTextWind.ClearWindow(Textcol, Backcol:ColType); BEGIN WindowCol:=(Backcol SHL 4)+Textcol; CurrentWindow; END; DESTRUCTOR ColTextWind.Done; BEGIN IF Open THEN CloseWindow; IF Saved<> NIL THEN Dispose(Saved); END; PROCEDURE IsCorM; BEGIN IF (MEM[0000:1040] AND 48) <> 48 THEN Scoff:= \$B800 ELSE Scoff:= \$B000; ScreenAdd:=PTR(Scoff,0); END; CONST MaxWind=10; VAR I,t:INTEGER; </pre>	<pre> Wind:ARRAY[1..MaxWind] of ColWindPtr; Tl:STRING(7); BEGIN RANDOMIZE; OS:=''; FOR I:= 1 TO 80 DO OS:=OS+' '; IsCorM; Tail:=NIL; Head:=NIL; NEW(Dummy); Heaperror:=@Heapfunc; FOR I:= 1 TO MaxWind do BEGIN STR(I:1,Tl); Wind[I]:=NEW(ColWindPtr, Tl,Tl.BorderType(RANDOM(2)+1),WHITE,BLACK)); IF Wind[I]=nil THEN BEGIN WRITELN('Halted- Initialisation Error'); HALT(1); END; END; TEXTBACKGROUND(0); TEXTCOLOR(15); CLRSCR; FOR I:= 1 TO 20 DO WRITELN('*****'); CopyScreenToDummy; FOR I:= 1 TO MaxWind do Wind[I]^^.OpenWindow(I*3,1,RANDOM(35)+5,RANDOM(5)+4); REPEAT REPEAT READLN(Tl); VAL(Tl,T,1); UNTIL T=0; IF (T>MaxWind) AND (T<=MaxWind*2) THEN BEGIN Wind[t-MaxWind]^^.ClearWindow(RANDOM(15), RANDOM(15)+1); END ELSE IF (T>0) AND (T<=MaxWind) THEN BEGIN Wind[t]^^.MoveWindow(RANDOM(36)+1, RANDOM(12)+1); END; UNTIL T=0; FOR I:= MaxWind DOWNTO 1 DO Dispose(Wind[I],Done); CopyDummyToScreen; END; </pre>
---	---	---

onto the end of the list, it automatically becomes the current window, and appears in front of all the others.

The program can be easily converted into a unit. There are a few points to watch when doing this.

- Ensure that you have enough heap RAM. My demonstration program uses ten windows, each taking just over 4000 bytes, plus another 4000 bytes for the shadow screen. This RAM must be obtained before you do anything else. Also, the Head and Tail pointers must be both preset to Nil.
- After clearing the main screen, remember to call CopyScreenToDummy, other-

wise it will look very messy when the shadow screen gets copied onto the screen.

It is probably safer to put some or all of this code into the unit initialisation section, so it gets called before the main program executes.

The Demonstration

Ten random sized windows are opened. Typing in a window number (1-10) will move the window to a new random position and have all I/O redirected there. Type in 11-20 and the window (number-10) will change colour and become current, staying in the same place but moving to the top. Type in 0 and it disposes of all of the win-

dows by calling the destructor Done for each. If a window was open, it is closed, and its storage space returned. Figure 3 summarises the use of the window objects.

During the development of this windowing system, I encountered a problem with assigning pointers. If you examine the code with a keen eye, you'll see the variable Otemp in ColTextWind.CloseWindow. Although the line Temp:=Temp^.Over is perfectly valid, it will not compile. After much hair pulling and chatting with Borland, I established that it was a compiler bug. It appears that the strict pointer type checking is just too strict. Otemp, which is declared as a pointer to ClassWindow at the same location as Temp, is the fudge that I used to get out of this. Thanks to Borland's technical support for confirming the bug and saving the rest of my hair!

EXE

David Bolton is 31, a graduate, and an experienced Turbo Pascal programmer. He runs BATPUG, the independent Turbo Pascal user group (write to 12 Clegg Avenue, Thornton, Cleveleys, Blackpool, LANCs FY3 1BJ for details of this group). His hobbies include Badminton, board and computer games and, in the last few weeks, getting married.

```

Var W1:ColWindPtr;

New(Dummy);
Head:=Nil;
Tail:=Nil;
CopyScreenToDummy;

W1:=NEW(ColWindPtr,Init(Title,Edge,Text Colour,Background Colour))

W1^.OpenWindow(Atx,Aty,LengthX,LengthY);

W1^.CloseWindow;

W1^.Done;

```

Figure 3 - Example code for Window objects

★ POCKET SIZED ★ PC-COMPATIBLE ★ PERSONAL ORGANISER ★



THE PC IN YOUR POCKET

The new Portfolio from Atari is the world's first pocket-sized electronic organiser that is also a powerful PC compatible computer, with a full QWERTY keyboard and scrolling 80x25 character display.

Portfolio includes everything you would expect from an electronic organiser - an address and telephone book, time manager, diary and sophisticated calculator.

Portfolio also includes several functions that you would not expect - a spreadsheet for your personal budget and expense records, as well as a text processor for typing memos and letters.

And, because Atari's Portfolio is PC compatible, it can communicate with your desktop PC at home, or in the office. You can transfer files from one to the other, to enable you to update your reports and figures with the Portfolio while you are on the move.

FOR ONLY **£249** INC VAT

SPECIFICATION

- ★ **Processor:** Intel 80C88 at 4.9152Mhz.
- ★ **Operating System:** Compatible with MS-DOS 2.11.
- ★ **Internal ROM:** 256K containing BIOS, operating system and applications suite.
- ★ **RAM:** 128K with an internal RAM disk, configurable from 8K. Externally expandable to 640K RAM.
- ★ **Keyboard:** 83 keys, QWERTY, IBM PC BIOS compatible. Burned numeric pad and function keys. Optional key click.
- ★ **Character Set:** Extended IBM ASCII (255 characters).
- ★ **Mass storage:** credit card sized memory cards (32K or 64K or 128K RAM).
- ★ **Display:** Graphics LCD, supertwist technology, MDA compatible, 40 columns x 8 lines, 240 x 64 pixels (with the option to window a full 80 x 25 character display). Keyboard controlled contrast.
- ★ **Peripherals:** 60 pin expansion BUS to take serial and parallel ports and memory expansion units.
- ★ **Size:** 8" x 4" x 1" (200mm x 105mm x 29mm).
- ★ **Weight:** 495 grammes (with batteries).
- ★ **Applications:** calendar and diary, address and phone book, Lotus 1-2-3 compatible spreadsheet, text processor, communications software.

COMING SOON!

In addition to the excellent software built-in to the Portfolio Free of Charge (see right) other software and peripheral products, such as the sophisticated Pocket Finance package and serial/centronics interfaces are available. And it doesn't stop there. Many manufacturers have recognised the potential of the Portfolio and have already started to design new peripherals and software. Products currently under development include: Serial interface with built in mini modem, Apple Macintosh interface, business, utility and programming software plus a range of adventure and battle strategy games. For further free details on the Portfolio range, fill in the coupon below and return it to Silica Systems now.

5 BUILT-IN APPLICATIONS! TO GET YOU ORGANISED!

To get you started, Portfolio comes with a suite of five useful functions built-in, all accessible from a simple menu display.

TIME MANAGER

More than just an electronic diary, the Portfolio Time Manager enables you to plan your appointments via a comprehensive calendar and diary. It can even be programmed to sound an audible alarm at specific times to remind you of important appointments.

CALCULATOR

Your Portfolio will be invaluable in the office or at home as a powerful pocket calculator. It has a full range of functions, including factorial,

power and root calculations, all with multi display formats and memories.

ADDRESS BOOK & DIALLER

Portfolio has a complete address book facility that allows you to store hundreds of addresses and phone numbers. And, at the touch of a button, you can retrieve any one of them, or search for a specific grouping, such as "all Italian restaurants". And, when you are ready to book your table, hold your Portfolio to your telephone mouthpiece and use its special built-in tone dialler to dial the number for you.

TEXT PROCESSOR

The Portfolio's built-in text processor program

includes word wrap, line and column count, string search, in fact most of the functions you would find in a word processor. It handles printer and word processor control codes and allows easy transfer of files between Portfolio and your desktop PC.

SPREADSHEET

For real calculating power, Portfolio has a Lotus 1-2-3 compatible spreadsheet built-in. It has 127 columns x 225 rows and reads/writes Lotus V1.0 and V2.01 files, so you can transfer data to and from Lotus 1-2-3 on your desktop PC. The Portfolio's 256K ROM includes MS-DOS and PC BIOS compatible systems software.



MEMORY CARDS

Portfolio can store and retrieve data and programs from its own RAM or from small credit card size memory cards that slot into its built-in card drive. The cards are available in three sizes, 32K, 64K and 128K, so you can carry a library of data in your pocket. The card drive also accepts ROM cards which can contain commercial or custom software.



POWER SUPPLY

Portfolio is powered by three AA batteries which will run for up to six weeks with normal use, or from the mains using an adaptor. All the peripherals take their power from the Portfolio, so no extra batteries or adaptors are required. A "battery-low" warning and memory back-up ensure that information is not lost when the batteries are changed.



INTERFACES & PERIPHERALS

Portfolio can communicate with other computers and supports a growing range of peripherals via a built-in 60 pin bus connector. Peripherals available include serial and parallel interfaces and memory expanders (to 640K). You can also load a card drive to your desktop PC, to enable it to read/write to Portfolio's cards.

SILICA SYSTEMS OFFER YOU

FREE OVERNIGHT COURIER DELIVERY: On all hardware orders shipped in the UK.
TECHNICAL SUPPORT HELPLINE: Team of Atari technical experts at your service.
PRICE MATCH: We normally match competitors on a "Same product - Same price" basis.
ESTABLISHED 12 YEARS: Proven track record in professional computer sales.
£13M TURNOVER (with 60 staff): Solid and reliable with maintained growth.
BUSINESS/EDUCATION/GOVERNMENT: Volume discounts available for large orders.
SHOWROOMS: Demonstration and training facilities at our London & Sidcup branches.
THE FULL STOCK RANGE: All of your Atari PC requirements from one supplier.
FREE CATALOGUES: Will be mailed to you with offers and software/peripheral details.
PAYMENT: By cash, cheque and all major credit cards.
CREDIT PAYMENT TERMS: Silica are licensed credit brokers - details on request.

Before you decide when to buy your new Atari Portfolio, we suggest you think very carefully about WHERE you buy it. Consider what it will be like a few months after buying your Portfolio, when you may require additional peripherals or software, or help and advice with your new purchase. And, will the company you buy from contact you with details of new Portfolio products? At Silica Systems, we ensure that you will have nothing to worry about. Silica Systems is a new division of Silica Shop, the UK's leading Atari specialists. This new division has been established to provide a service to the more serious home user, as well as to business and education purchasers.

Silica have been established for over 12 years, and have an annual turnover of £13 million. With our unrivalled experience and expertise, we can now claim to meet our customers' requirements with an understanding which is second to none. But don't just take our word for it. Complete and return the coupon now, for our latest Free literature and begin to experience the "Silica Systems Service".

**SILICA
SYSTEMS**



MAIL ORDER: 1-4 The Mews, Hatherley Rd, Sidcup, Kent, DA14 4DX Tel: 081-309 1111
Order Lines Open: Mon-Sat 9:00am-6:00pm No Late Night Opening Fax No: 081-308 0608

LONDON SHOP: 52 Tottenham Court Road, London, W1P 0BA Tel: 071-580 4000
Opening Hours: Mon-Sat 9:30am-6:00pm Late Night: Thursday until 8pm Fax No: 071-323 4737

SIDCUP SHOP: 1-4 The Mews, Hatherley Rd, Sidcup, Kent, DA14 4DX Tel: 081-302 8811
Opening Hours: Mon-Sat 9:00am-5:30pm Late Night: Friday until 7pm Fax No: 081-309 0017

BUSINESS/EDUCATION: 1-4 The Mews, Hatherley Rd, Sidcup, Kent, DA14 4DX Tel: 081-308 0888
Order Lines Open: Mon-Fri 9:00am-6:00pm Closed on Saturdays Fax No: 081-308 0608

To: Silica Systems, Dept EXE-0690-29, 1-4 The Mews, Hatherley Rd, Sidcup, Kent, DA14 4DX

PLEASE SEND INFORMATION ON PORTFOLIO

Mr/Mrs/Ms: Initials: Surname:

Address:

Postcode: EXE 7 9D

Tel (Home): Tel (Work):

Company Name:

Which computer(s), if any, do you own?

£80E - Advertised prices and specifications may change - please return the coupon for the latest information.

CIRCLE NO. 084

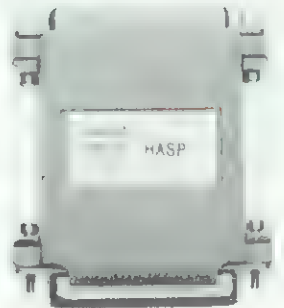
The HASP Family: Software protection

Aladdin Knowledge Systems is a leading company in the field of software protection. During the last five years they have enabled hundreds of producers in more than 30 countries to protect their software.

They believe that software protection is a serious business which demands a high level of knowledge and expertise in order to give producers a reliable and lasting protection for their intellectual property. Based on research and experience in the software protection field Aladdin have now added two new dongles to the already widely used **HASP II** system, the new additions are **HASP 3** and **MEMOHASP** with its read/write memory.

All the systems attach to the parallel printerport of the IBM and compatible PC/XT/AT and PS/2 families of personal computers. To install **HASP**, you can either use the installation program on an EXE or COM file, the protected program will then run only if a **HASP** plug with its own unique code is attached to the computer. A second option is to link an object file to the software, which allows the developer to decide how he wishes to respond if the expected code is not returned. Finally it is possible to use a resident program provided in the software which is supplied with the plug. All the installations procedures are explained in this software, thus making protecting your programs a simple operation.

Some advantages of the **HASP** system are that the plug can be connected and disconnected in a matter of seconds, it is transparent to the operation of the protected software, to the operating system and to the printer.



For further details contact Adrian Harris at:

EM **EMPIRIC LTD**

IPS House, 5 Bridge Avenue, Maidenhead, Berks. SL6 1RR
Telex: 849462 TELFAC G Facsimile: (0628) 784522
Tel: Maidenhead (0628) 35913 International: +44 628 35913

CIRCLE NO. 112

FTN77/486

Ready when you are!

- ☐ The FASTEST PC Fortran compiler around.
- ☐ Compiled programs exploit the 486 hardware using paged virtual memory giving an address space of up to 4 Gigabytes.
- ☐ ANSI Fortran 77 plus mainframe extensions.
- ☐ Runs under MS-DOS. FTN77/486 is supplied complete with our DBOS/486 DOS extender.
- ☐ Optional global optimisation or checkout mode with diagnostics.
- ☐ Full window-based source level debugger.
- ☐ Provides a library of over a hundred routines for MS-DOS access, non-standard file manipulation, window management, graphics, etc.
- ☐ Supported by experts and supplied with a comprehensive, fully indexed reference manual.
- ☐ Allows libraries to be linked dynamically to programs during execution. Dynamic linking means that EXE files are kept small.
- ☐ Contains a full 32-bit in-line symbolic assembler with window-based debugger.
- ☐ Interface to real mode libraries.
- ☐ Selection of graphics libraries available.
- ☐ Optional Weitek co-processor support.

FTN77/386 AVAILABLE NOW!

SALFORD SOFTWARE MARKETING LTD

COMPUTING CENTRE · UNIVERSITY OF SALFORD · SALFORD · M5 4WT · ENGLAND
Telephone (+44) 061 745 5678 Telefax (+44) 061 745 5666

*MS-DOS is a trademark of Microsoft Corporation. IBM and PS/2 are trademarks of the International Business Machines Corporation. DBOS/486, FTN77/386, FTN77/486 are trademarks of the University of Salford

CIRCLE NO. 085

THE ATARI ABC

Now, there's a PC-AT compatible that not only solves problems like other AT compatibles, it also solves the one problem that its predecessors have created ... affordability.

Silica Systems are pleased to present the new Atari ABC. The ABC is a 286 AT compatible that runs at over 5 times the performance of the IBM XT. This is achieved by combining the power and speed of a 16-bit 286 processor, with Atari Corp's development experience and engineering capabilities, which use the very latest in design technology. However, the Atari ABC's are available at 'XT prices' and many XT owners will wonder why they paid more, but got less!

The ABC is a reliable, high performance computer, built to exceptionally rigorous standards. It is well designed, to a state of the art specification, maintaining maximum expansion capability for the future. This includes up to 4Mb of RAM and 3 AT expansion slots. Plus, unusually for a PC at this price, the ABC has the ability to install two or three extra drives, with the cabling already inside the CPU.

The ABC has a host of impressive features, all built-in as standard, encouraging simple installation and ease of use. In addition, the ABC's small footprint and quiet operating, mean that you will notice it less on your desk than other PC's. Except, of course, when you begin to take advantage of its AT power.

At its remarkably low price point, the ABC is ideally suited for home, office and educational use and will take full advantage of the vast range of PC compatible software. And, with 12 months FREE on-site maintenance (8 hour response), you can rest assured that your ABC will be fully supported.

The ABC is the latest addition to a comprehensive range of both 286 (8Mhz or 12Mhz) and 386 (16Mhz and 20Mhz) PC's, from the Business Systems Division of Atari Corp. Atari's PC range offers an unsurpassed combination of reliability, compatibility and expandability, with unique features like Atari's 44Mb removable hard disk on their PC4. Return the coupon NOW for further details of the complete range of PC products from Atari's Business Systems Division.

**ABC
TURBO**

AVAILABLE, JUNE 90

To put the ABC even further ahead of the competition, we have introduced a special 'Turbo' version, which includes a super fast RLL hard disk controller to increase the data transfer rate from 150KBytes/s to 800KBytes/s, with a performance index of 4.801. Plus to provide exceptional graphics with an outstanding colour resolution of up to 1024x768, we have included a Super VGA graphics adaptor and a 14" VGA colour monitor (which also supports 1024x768 resolution), all for an additional price of only £300. Check the ABC Turbo against the competition, at only £1299 (+VAT) with 30Mb hard disk, Super VGA colour graphics capabilities, plus 12 months on-site maintenance, it offers unbeatable value.

286 AT

WITH 30Mb HARD DISK & EGA MONITOR - £899

FREE! - ON-SITE MAINTENANCE

FREE! - TECHNICAL SUPPORT HOTLINE

FREE! - TEST/BURN-IN BEFORE DESPATCH

FREE! - OVERNIGHT COURIER DELIVERY

£899
WITH +VAT
**30Mb
HARD
DISK
AND
EGA
MONITOR**



**ABC
TURBO
£1299**
EXC VAT

• SVGA COLOUR MONITOR
• SVGA GRAPHICS CARD - 512K
• 30Mb HARD DISK DRIVE
• FAST DISK CONTROLLER

£899

+VAT=
£1033.85

ATARI

NO MONITOR

+ EGA MONO

+ EGA COLOUR

**ABC
TURBO**
AVAIL. JUNE 90

• 14" SVGA COLOUR MONITOR
• SVGA GRAPHICS ADAPTOR
• FAST DISK CONTROLLER

	3 1/2" FLOPPY DRIVE NO HARD DRIVE	3 1/2" FLOPPY DRIVE + 30Mb HARD DRIVE
NO MONITOR	£599 +VAT = £668.85	£799 +VAT = £818.85
+ EGA MONO	£699 +VAT = £803.85	£899 +VAT = £1033.85
+ EGA COLOUR	£799 +VAT = £918.85	£999 +VAT = £1148.85
ABC TURBO	£1099 +VAT = £1263.85	£1299 +VAT = £1493.85

LOW PRICE - HIGH SPEC!

CPU + MOTHERBOARD	<ul style="list-style-type: none"> 8Mhz 80286 16-bit CPU Landmark Speed = 10.3Mhz, Norton SI Test = 9.0 Latest NEAT (New Enhanced AT) technology chip set Has user selectable bus, DMA speeds & wait states Programmable to operate at 0 or 1 wait states 640K RAM base memory (expandable) Motherboard expandable to 4Mb RAM memory Real time clock with battery back-up Full 16-bit data line capability Socket for 80287-8 numeric Co-Processor 3 16-bit AT expansion slots (max)
DISK	<ul style="list-style-type: none"> 32Mb RLL hard disk - 150KBytes/s transfer - 60ms access Fast disk controller - 800KBytes/s transfer - Turbo version 1.44Mb built-in 3 1/2" floppy disk drive Space for two 5 1/4" and two 3 1/2" drives (max)
I/O	<ul style="list-style-type: none"> Parallel port built-in (25-pin Centronics) Serial port built-in (9-pin RS-232) Mouse controller port built-in (9-pin serial)
GRAPHICS	<ul style="list-style-type: none"> Super VGA video graphics controller - 512K - Turbo vers Super VGA: 256 colours in 800x600 res - Turbo vers EGA video graphics controller built-in EGA: 16 colours available from a palette of 64 EGA: 256K of dedicated Video RAM built-in MDA, CGA and Hercules graphics controller built-in
PHYSICAL	<ul style="list-style-type: none"> Low noise operation Ergonomic small footprint (38cm x 41cm x 15cm) Keyboard - 102 key enhanced AT (12 function keys) Keyboard - has 2 position height adjusters
PLUS!	<ul style="list-style-type: none"> Fully OS/2 compatible (requires 4Mb RAM) Supports Industry Standard Networks MS-DOS 3.3 with GW BASIC 3.22 12 Months FREE on-site maintenance included

SILICA SYSTEMS OFFER YOU

ON-SITE MAINTENANCE (8 hour response): Available Free of Charge on all ABC's.
TECHNICAL SUPPORT HELPLINE: Team of Atari technical experts at your service.
100% QUALITY CONTROL: Every ABC is tested/burnt-in before despatch.
FREE OVERNIGHT COURIER DELIVERY: On all hardware orders shipped in the UK.
ESTABLISHED 12 YEARS: Proven track record in professional computer sales.
£13M TURNOVER (with 60 staff): Solid and reliable with maintained growth.
BUSINESS/EDUCATION/GOVERNMENT: Volume discounts available for large orders.
PRICE MATCH: We normally match competitors on a "Same product - Same price" basis.
SHOWROOMS: Demonstration and training facilities at our London & Sidcup branches.
THE FULL STOCK RANGE: All of your Atari PC requirements from one supplier.
FREE CATALOGUES: Will be mailed to you with offers and software/peripheral details.
PAYMENT: Cash/cheque/credit card/credit - we are licensed credit brokers - details on request.

Before you decide when to buy your new Atari ABC, we suggest you think very carefully about WHERE you buy it. Consider what it will be like a few months after buying your ABC, when you may require additional peripherals or software, or help and advice with your new purchase. And, will the company you buy from contact you with details of new ABC products? At Silica Systems, we ensure that you will have nothing to worry about. Silica Systems is a new division of Silica Shop, the UK's leading Atari specialists. This new division has been established to provide a service to the more serious home user, as well as to business and education purchasers. Silica have been established for over 12 years, and have an annual turnover of £13 million. With our unrivalled experience and expertise, we can now claim to meet our customers requirements with an understanding which is second to none. But don't just take our word for it. Complete and return the coupon now, for our latest Free literature and begin to experience the "Silica Systems Service".

**SILICA
SYSTEMS**



CIRCLE NO. 086

MAIL ORDER: 1-4 The Mews, Hatherley Rd, Sidcup, Kent, DA14 4DX Tel: 081-309 1111
Order Lines Open: Mon-Sat 9.00am-6.00pm No Late Night Opening Fax No: 081-308 0008

LONDON SHOP: 52 Tottenham Court Road, London, W1P 0BA Tel: 071-580 4000
Opening Hours: Mon-Sat 9.30am-6.00pm Late Night Thursday until 8pm Fax No: 071-323 4737

SIDCUP SHOP: 1-4 The Mews, Hatherley Rd, Sidcup, Kent, DA14 4DX Tel: 081-302 8811
Opening Hours: Mon-Sat 9.00am-5.30pm Late Night Friday until 7pm Fax No: 081-309 0017

BUSINESS/EDUCATION: 1-4 The Mews, Hatherley Rd, Sidcup, Kent, DA14 4DX Tel: 081-308 0888
Order Lines Open: Mon-Fri 9.00am-6.00pm Closed on Saturdays Fax No: 081-308 0008

To: Silica Systems, Dept EXE-0690-28, 1-4 The Mews, Hatherley Rd, Sidcup, Kent, DA14 4DX

PLEASE SEND INFORMATION ON ATARI PC's

Mr/Mrs/Ms: Initials: Surname:

Address:

Postcode:

Tel (Home): Tel (Work):

Company Name:

Which computer(s), if any, do you own?

£80E - Advertised prices and specifications may change - please return the coupon for the latest information.



The definitive database for micros and micro-networks.

MAIN FEATURES

- Clear Concise English-like programming language.
- Rapid system development and easy maintenance.
- Relational and Hierarchical structures.
- Variable length records.
- Powerful facilities for reading and writing files from other systems - e.g. Pegasus, Lotus 123.
- Run-time licences available.

OS/2, PC-DOS, MS-DOS, XENIX,
CDOS, MS-NET, NOVELL ETC.

Microft Technology Limited
The Old Powerhouse, Kew Gardens Station
Kew, Surrey TW9 3PS.
Telephone: 081-948 8255

£560+VAT Single User.
£950+VAT Network Version up to 5 Users.
£100+VAT each subsequent User.

CIRCLE NO. 094

USING INFORMIX?

The E.I.U.G. is an independent non-profit making group of Informix users which acts as a forum for the exchange of information, ideas and experiences. We publish an informative newsletter and hold regular meetings for members. Our next meeting is in October 1990 in central London.

For further details contact

Bryce Campbell on 0532 738244
or Jeremy Russell on 081 446 6481

EUROPEAN INFORMIX USERS GROUP

Informix is a trademark of Informix Software Inc.

CIRCLE NO. 116

SMOOTH SKATING BETWEEN DOS AND UNIX/XENIX

*** ICE.TEN ***

For those who routinely trek between DOS and UNIX, ICE.TEN is the "system in middle". It's everything you need to fuse DOS and UNIX.

- Instantaneous switching between DOS and UNIX.
- WYSE 60, ANSI, VT220 & VT100 terminal emulations.
 - Support for Speeds up to 115,000 kbaud.
 - Supports ANSI colour codes in all emulations.
 - Fully programmable function keys.
- Transparent and local print to PC's parallel port.
 - Easy file transfer between PC and Host.
- DOS applications can share UNIX printers.
- Use DOS commands on your UNIX host.
- Multi PC license gives incredible value.
- Full modem support with automatic dial facility.

For details call Stuart Hobbs on 071-323 2297

Custom Business Systems Ltd
MSDOS to UNIX EXPERTS

CIRCLE NO. 089

G.W. Computers Inc.
ADC1, 4 Eagle Square,
E.Boston, MA02128
U.S.A.

Tel: 617-569-5990
Fax: 617-567-2981

System-IX (Networks) Ltd.
55, Bedford Court Mansions,
Bedford Avenue, London,
WC1B 3AD U.K.

Tel: 071-636-8210
Fax: 071-255-1038

Menu-IX System-IX DBMS-IX System AI Spell Merge Lock GURU Crypt Task Target



Hardware System
386-16 SX starting from £895.00 to
486-25 w: 4mb ram, VGA, .80mb disk £3750.00
call for quotations and latest prices.



SCO XENIX		286	386
SCO Xenix Op Sys		£357	£422
SCO Xenix Dev Sys		£357	£455
SCO Xenix Text Pro		£126	
SCO Xenix Comp Sys		£775	£905
SCO Unix Op Sys		£525	
SCO Unix Dev Sys			£613
SCO CUI Graph Run/Sys		>	£191
SCO Multiview		£256	£321
SCO VFile			£321
SCO Foxbase+		£516	£647
SCO Foxbase+ Run/Sys		>	£191
SCO Masterplan		>	£516
M/Soft Word v.3.0		>	£563
Multiplex		>	£321
SCO Ercoff Documentation		£651	£516
SCO Lyrinx v6.0		>	£386
SCO Professional		>	£386
SCO Image Builder		>	£451
M/Soft BASIC compiler		>	£256
M/Soft BASIC interp.		>	£451
M/Soft FORTRAN comp.		>	£451
M/Soft PASCAL comp.		>	£451
SCO XENIX-NET Server		£386	£451
SCO XENIX-NET 3 Server		£841	£1096
SCO XENIX (Rev 02-01) Run/Sys.			£227

OPEN DESKTOP and XNIGHT also available. Call
Complete systems (hardware) include: MENUIX FREE!

INTERACTIVE 386/ix UNIX		1-2 Users	Multi User
New Rel 2.2	Application Network Workstation		
Platform (2 User)		£495	£345
(Multi)		£925	£585
Developer (2 User)		£945	£1295
(Multi)		£1359	£1460

These are VERY powerful professional systems. For an Application starter you need about 12-20 mb of hard disk, but a workstation developer needs hard disk space to install 65 by 1.2 or 1.44 mb diskettes. To include things like: Op Sys, Softdev (LEX, YACC, SOCS etc), Standard C, New C, Text per (NROFF, TROFF, TBL etc), XIL, XII Dev, Vpix (DOS SHELL), TCP/IP, NFS, Online MANS, Looking glass, Visman UI, Codewatch, Co-Edit. Full documentation weighs in at about 40 lbs. All modules can be purchased separately.

Some Parts		1-2 Users	Multi User
386/ix Runtime		£255	£495
Software Development System			£495
X11.3 Runtime		£240	£265
X11.3 Development System		£580	
Multiview			£395
Mod/Development System		£365	£355
TCP/IP			£420
NFS		£355	£385
TCP/IP + NFS		£425	£385
PC Interface, DOS Bridge Module			£365

System kit over £300 include MENUIX FREE.
System kit over £1000 include SYLIX and GURU FREE!

Microport UNIX		286	386
Complete System		£395	£495
Comp. Sys. Unlimited		£538	£718
Runtime System (2 user)		£159	£189
Run. Sys. Unlimited		£299	£358
Unlimited User Upgrade		£159	£189
Software Development	£189	£329	
Text Preparation		£129	£129
DOS Merge Limited		£165	£249
DOS Merge Unlimited		£165	

Additional HOT Items		
Q-Calc spreadsheet		£695
Master-Plan project management		£1295
Uniplex business software		£1150
PC-CONNECT		£450
SQL-Connect		£450
PC-TCP		£375
PC-Xview		£245
Xvision		£295
Emacs		£325
Designer C++		£1195
Terminal Control for UNIX AND DOS		£595
Manipix		£49.95
Synix		£99.95
Synix + Guro		£145
System-IX		£295
3-rod-IX		£295
Chrit-IX for DOS		£50

We Stock Hundreds Of DOS Products From
ALDUS To ZSOFT With LOTUS,
MICROSOFT & SAGESOFT In Between!

Exceptional products: Entire Shakespeare works,
Thesaurus & Dictionary, Bibles, 500,000 word
lists, Moby Words, CD-ROMS, C sources, etc.

We also sell CD-ROM drives and WORM drives,
Network cards, specialist printers and plotters,
Laser printers and colour thermal printers (QMS).

*** Brand NEW ***

Portable hard disk technology. FITS IN A STANDARD 5.25" DRIVE BAY.
FRAME & REMOVEABLE DRAWER, enables you to fit a 3.5" drive of 40-200 mbytes
into the draw, and remove it in 1 second for safeguard or even better, to just plug in any
O/S of your choice, or take your file-systems from machine to machine. JUST £55

Compaq Toshiba IBM Epson Qume NRC Seagate Wyse Reliance Smith Prism Renner

CIRCLE NO. 087

A Real Time Pioneer

They don't build 'em like this any more. You'd better hurry, or you'll miss your last chance to see working examples of a venerable old programming language, as Jim Cooling explains.

As time goes by, new programming languages are introduced, some are improved, some just fade away. So most experienced software designers end up with a good working knowledge of a number of languages, working (hopefully) with state-of-the-art techniques. Many programmers retain a soft spot for a particular language which they met along their career path. For me that language is CORAL 66, Computer On-line Real-time Application Language.

CORAL 66 has its roots in the UK defence industry. It is still the most widely used high-level programming language in that area, although eventually it will succumb to the Ada steamroller. Even after 24 years, its low-level facilities (in terms of ease of use, visibility, clarity and power) are matched by few of its successors.

What were the objectives of its designers? It's probably best to stress what they didn't set out to do: ie to rival COBOL in the business world or FORTRAN for number crunching. CORAL 66 was designed to be used as a general purpose programming language for real-time embedded systems. The intended targets were small to medium sized dedicated computers, frequently being applied to control applications (it was, for instance, widely used on the Ferranti Argus minicomputer in ship-board Radar and Weapon systems). Further, it was developed to support both fixed point and floating point arithmetic for such real-time functions. And, most importantly, its primary objective was to reduce software costs.

How was this reduced cost to be achieved? The designers considered that the increased productivity attainable by using a

high-level programming language would lead to:

- Lower programming costs.
- Faster implementation of designs.
- Simplified and reduced maintenance efforts.

It was also realised that these goals couldn't be attained without suffering drawbacks. Compared with assembly language programs, CORAL 66 programs would need more store space and would take longer to execute. At that time, the primary store in computers was magnetic core, which was heavy, large and slow. Even so, it was deemed that the advantages far outweighed the disadvantages. Given the circumstances, this was a brave decision.

The language's design roots were in JOVIAL, a language developed by the United States Air Force in 1959. The other major influence was ALGOL 60, 1960. Clearly it was going to be a block structured language, though some features of FORTRAN were incorporated.

Program Layout

The simplest type of CORAL 66 program consists of a single compilable unit, structured as shown in Figure 1. A block section is defined using the predefined ('basic') words BEGIN and END. Program declarations are made within the block, restricting the scope of declared items to that block. Basic words are enclosed in single quote marks. The official definition of the language recommended that such words should be written in upper case, all others being in lower case - the form shown here was permitted as an alternative. However, at the time that language compilers began to be produced in earnest, data terminals which could handle both cases were quite expensive. Hence many compilers allowed the programmer to write source code using the single case form (more precisely, they

1962 - Original definition of CORAL

1965 - The Royal Radar Establishment (RRE) began the project by producing an initial definition of CORAL. This, later formalised as CORAL 64, was a subset of JOVIAL.

1966 - CORAL 66 was devised by IF Currie and M Griffiths, aimed for applications running on minicomputers.

1967 to 1969 - Compilers were produced for a number of machines, including the Marconi Myriad and the Plessey XL4.

1970 - CORAL 66 was specified as the preferred language for UK defence projects. The official definition of the language, written by PM Woodard, PR Wetherall and B Gorman, was published by HMSO.

1973 - As a result of discussions between the Department of Industry and various manufacturers and trade associations, a CORAL 66 support organisation was formed.

- Technical support - RRE.
- Promotion, education and information - National Computing Centre.

1974 - Major conference: 'CORAL 66 The real-time language for current use'.

1980 - Standardised by the British Standards Institute as BS5905:1980.

POWER

AND



FLEXIBILITY

BRIEF 3.0 increases your productivity for just £159.00*

*VAT extra, FREE delivery

BRIEF has established itself as the leading programme editor on the market today. Now, with the launch of the latest enhanced version - BRIEF 3.0 - it offers even better productivity.

ENORMOUS UNDO POWER

BRIEF 3.0's enhanced UNDO capability takes the fear out of programming. BRIEF 3.0 allows you to undo any command that affects the cursor or the text (including global changes) up to 300 times, even back to the beginning of your session. It's the safety net you need to program with confidence.

REAL FLEXIBILITY

There's built-in support for 36 languages including Assembler, BASIC, C, COBOL, dBase, FORTRAN and Pascal. And thanks to its complete macro language you can make BRIEF 3.0 fit your own programming style with ease: change key assignment, commands, prompts, colours - even the indenting styles.

BRIEF 3.0 - new benefits

- With our new C-like macro language, CBRIEF, C programmers can write macros on the fly and not have to switch between C and a different macro language
- The new source level debugger works with both the original BRIEF syntax and the new CBRIEF macro language
- Maximum lines per file has been increased to 4 billion
- We've added to and improved template editing and smart indenting for ADA, Cobol, Basic, Fortran, Modula-2, Pascal and C
- We've added editable, multiple keystroke macros that can be saved and restored
- You can even save and restore your window set-ups

Plus the features that made BRIEF famous

- Flexible windowing
- Unlimited number of files
- File size limited only by disk space
- Regular expression search and replace

No other editor puts this much power and flexibility in your hands. So if you're looking for extra productivity purchase your copy of BRIEF 3.0 today.

Solution Systems

SOLUTION SYSTEMS are the UK distributors for BRIEF 3.0. When you buy from us you get the authorised UK version of BRIEF 3.0 dual media (3½" and 5¼") plus bonus macro diskette.

Special phone line for Access/Visa card holders.
Save time by phoning
0763 244141
between 9.00 and 6.00
Monday to Friday
to place your order

BRIEF 3.0 The Programmer's Editor

Send to: Solution Systems UK, The Maltings, Green Drift, Royston, Herts SG8 5DB

Please rush me my copy of BRIEF 3.0

☐ I enclose a cheque to the value of £182.85 (£159.00 + V.A.T.) crossed and made payable to Solution Systems UK.

Orders despatched same day.

NAME

ADDRESS

POSTCODE

☐ Please charge my Access/Visa* account the sum of £182.85 (£159.00 + V.A.T.)

Card No

Expiry Date

Signature

*Delete as appropriate

CIRCLE NO. 090

EXE 7-90

were generally case insensitive).

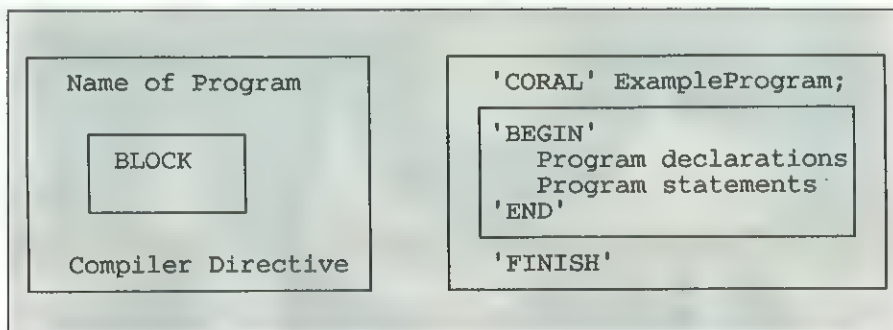
The smallest unit which can be compiled on its own is called a segment (nothing to do with 80x86 segments) delimited by the basic words BEGIN and END. Most well-designed practical programs consist of a number of segments, providing the programmer with extensive information hiding facilities. Segments can be compiled in separate files, then linked together. There are various facilities to allow communication between separately linked modules: COMMON (to access another part of the application, in another segment), LIBRARY and EXTERNAL (to get at library procedures and operating system primitives - LIBRARY is frequently used as an include directive) and ABSOLUTE (to get at an absolute address). These directives are placed at the top of the code module. Difficulties may be met when using communicators, as only COMMON is fully defined; the functions of the others depend on the compiler implementation.

Structuring Facilities

How well does CORAL 66 support the aims of structured software design? This is an interesting question, as the CORAL 66's language design pre-dates much of the work on structured techniques. What we are looking for in the language are facilities to:

- Build structured programs.
- Implement entities within this (entity is rather an overworked word - perhaps thingy would be better).

Figure 1 - Simplest form of Coral 66 program



- Manipulate and control these entities - program control.

***CORAL 66 is still
the most widely
used language,
although
eventually it will
succumb to the
ADA steamroller***

Modern structured software designs rely heavily on modular program construction and virtual (abstract) software machines.

Segmentation, together with block structuring, gives us a simple and straightforward way of building modules. The virtual machine is implemented using the procedure construct, these coming in three forms: the procedure, the typed procedure and the recursive procedure.

The procedure structure consists of two parts, a header

'PROCEDURE' Proc(Params);
and a body. Both value and reference (called 'location') parameters may be used. Typed procedures, defined with a header NumType 'PROCEDURE'

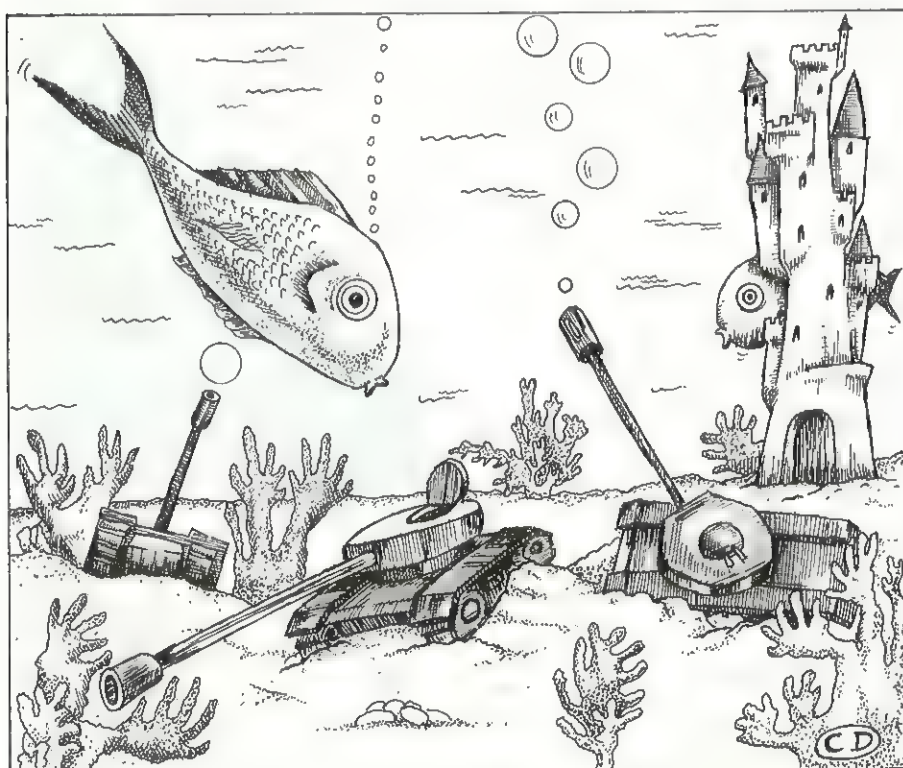
Proc(Params);
are analogous to Pascal's functions (NumType is 'INTEGER', 'FIXED' etc); they are used where a result is to be passed back from the procedure. The body of this procedure type must contain an 'ANSWER' statement to return the required value. The features of these procedure types should be obvious. The third type, the recursive procedure, differs from the others only in that it is declared using the 'RECURSIVE' keyword and it can invoke itself. In CORAL 66, special code is needed to implement recursion. The high data storage space overhead of recursion makes it an expensive luxury in small embedded systems, so the language designers chose to highlight its use with this mechanism.

Data types fall into two groups: simple (or scalar) and structured. CORAL 66 supports both constructs, but only in a very limited way. Everything is defined in terms of its number representation within the computer, thus the simple types are 'INTEGER', 'FIXED' and 'FLOATING'. Data abstraction is attained by using named variables

```
'INTEGER' SkinColour,  
UartStatus;
```

This typing is rather weak - the compiler can check only for structural, and not name, equivalence. For instance, the assignment statement

```
SkinColour := UartStatus;
```



DON'T GIVE IT AWAY!

Software Developers! Protect Your Software Against Unlicensed Use.

From **Link Computer Denmark**, a remarkable piece of pure software that will lock a DOS .com or .exe file to a diskette, machine or network so that you may copy (or back-up) to your heart's delight, but you can still only run the program on the designated medium.

It's called **Cop's Copylock II** and it's clever stuff. You get an unlimited number of protections per purchase and its also uncrackable. Forget your favourite copying programs - they won't work on this. Forget Option Board and Bit Copiers - we dealt with them years ago.

If you don't want to dabble with dongles and aren't so keen on your software being freeware, you ought to take a look at this.

Ring **Dino Cano** on
Tel: 0763 263 073 Fax: 0763 262 620

Link Computer ApS.

CIRCLE NO. 091

CopyControl

THE NEW GENERATION OF COPY PROTECTION

- NO** User hassle
- NO** Back-up problems
- NO** Hardware add-ons or special disks needed
- NO** Changes to source code required
- YES** CopyControl beats ALL bit-copier Programmers
- YES** Floppy disks, hard disks and networks supported
- YES** CopyControl is totally transparent to the user
- YES** You can limit program use by no. of copies, no. of executions or date
- YES** CopyControl works on all IBM compatibles
- YES** Free demo disk available

ONLY £419 (+£5 p&p + VAT)

For Further Information Phone or Write to:

microcosm

Microcosm Limited
17 Cranbrook Road, Bristol BS6 7BL
Telephone: 0272-441230

CIRCLE NO. 092

OOPS + AI -> Prolog⁺⁺

Now you can combine the best features of AI and OOPS with LPA's new Prolog⁺⁺, the all new object-oriented extension to Prolog.

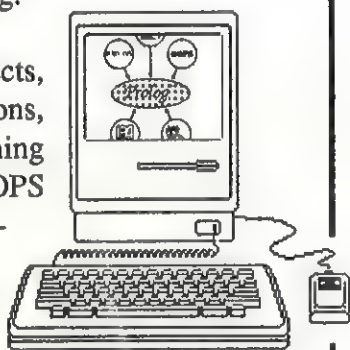
Prolog⁺⁺ integrates objects, methods, messages, functions, attributes - in short, everything you would expect in an OOPS environment - with the declarative power of Prolog.

All of the standard features of LPA Prolog, such as source level debugging, incremental compilation, sophisticated graphics and windows, and so on, are inherited by Prolog⁺⁺.

So why not be objective, and give us a call?



Logic Programming Associates Ltd
Studio 4, RVPB, Trinity Road, London,
SW18 3SX, England
Phone: 081 871 2016
Fax: 081 874 0449



Humanize your Computer!

Lets face it, your computer can be pretty dumb. It just sits there, crunching numbers and displaying text and graphics. Does it think about what it's doing? Does it, heck!

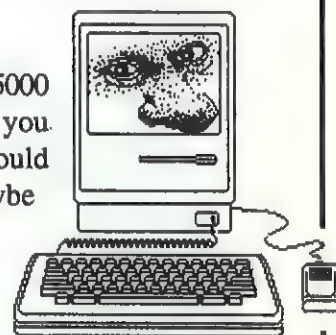
Suppose it tells you to buy \$5000 of ACME shares, wouldn't you like to know why? If you could *humanize* the machine, maybe it could just tell you!

That's just why we at LPA developed *flex*: an object oriented, frame based Expert Systems toolkit. It gives your computer the power to think.

To find out how to humanize *your* PC or Mac, why not give us a call?



Logic Programming Associates Ltd
Studio 4, RVPB, Trinity Road, London,
SW18 3SX, England
Phone: +44 1 871 2016
Fax: +44 1 874 0449



CIRCLE NO. 117

CIRCLE NO. 118

Figure 2 - Coral 66 control structures

CONDITIONAL CONSTRUCT	CONDITIONAL EXPRESSION
Selection	'IF'...'THEN'...;
Iteration - pre-check loop	'IF'...'THEN'...'ELSE'...;
- post-check loop	'FOR'...'WHILE'...'DO'...;
- fixed looping	'FOR'...'STEP'...'UNTIL'...'DO'...;
	'FOR'...'STEP'...'UNTIL'...'DO'...;

will be accepted as perfectly valid - even if it's nonsense in program terms.

The structured types are the 'ARRAY' and the 'TABLE', but their elements must belong to one of the simple types. CORAL 66 Arrays are one dimensional types, needing little further explanation. The table is similar to a two dimensional array, but having special qualities to allow very efficient packing of data. Moreover, it allows variables of different types to be stored together (in that way it is like the record structure).

CORAL 66 provides a set of control structures which conform to the rules of structured programming: sequence, selection and iteration. These are shown in Figure 2. Program control can also be exercised by the use of labels and GOTO statements (normally considered bad practice, but ideal for use with exception handlers). Labels are normally individual and distinct.

However, one construct, the switch, treats its label as if it is actually a set of labels. For example:

```
'SWITCH' PressureAlarms =
    ExtraLow, Low,
    High, ExtraHigh;
the statement 'GOTO' PressureAlarms[0] transfers control to label ExtraLow and 'GOTO' PressureAlarms[3] transfers control to label High. Using this, a crude form of the case selector mechanism can be built.
```

Most modern programs are built on the concepts of functional structuring of designs, incorporating top-down decomposition, stepwise refinement of the problem, and the use of structured programming within the source code (in spite of the hype, these ideas carry straight across into object oriented design). These are relatively easy to implement in CORAL 66.

Low-level

In the real time embedded world, computer software and its environment are inseparable items. To build effective and efficient programs we need to access and manipulate system objects (actual and virtual) easily and simply. This involves device access and control, bit manipulation and assembly language interfacing.

For example, suppose that our system has a serial I/O channel, driven by a memory-mapped UART. This device can be named, and its address location declared, in a simple manner:

```
'ABSOLUTE' ('BYTE'
Uart1Data/'HEX' (A005),
Uart1Control/'HEX' (A006));
```

The variable Uart1Data is located at absolute address A005H, its data unit size being a byte. Operations with this variable are straightforward. Very efficient coding can be achieved by using constructs such as:

```
Uart1Control := 'HEX' (40);
```

The programmer can get very close to the object code without having to resort to assembly language coding.

When dealing with hardware devices we often have to interpret and control individual bits (or groups of bits). This is handled by the 'BITS' operator, which works like this:

```
'BITS' [NoOfBitsAccessed, StartPos]
VariableName.
```

This is a very flexible tool, for instance, you can legally write

```
'BITS' [4,0]DacWord :=
'BITS' [4,8]ControlSignal;
```

Working at this level, we invariably find ourselves having to make logical decisions. The logical operators of CORAL 66, 'DIFFER', 'UNION' and 'MASK', equivalent to Boolean 'exclusive or', 'inclusive-or' and 'and' functions, fulfil this purpose. Nonetheless, it remains almost impossible to develop software for real-time embedded systems without using assembly language programming at some stage. This can be done in CORAL 66 using code inserts. This example shows the saving of register contents in an 8085 processor:

```
'CODE'
'BEGIN'
DI
PUSH PSW
PUSH B
PUSH D
PUSH H
SIZ 5
'END';
```

```
'CORAL' TRIANGLE;
(Program to solve triangle problem in CORAL 66)
'DEFINE' SIDES PER TRIANGLE "3";
'DEFINE' PRINT TEXT "PRTTXT";
'DEFINE' GET INTEGER "GETINT";
'DEFINE' PRINT INTEGER "PRTINT";

'COMMENT'
***** External Procedures *****
'EXTERNAL' {
'PROCEDURE' PRINT TEXT ('VALUE' 'INTEGER' );
'PROCEDURE' GET INTEGER ('VALUE' 'INTEGER' );
'PROCEDURE' PRINT INTEGER ('VALUE' 'INTEGER', 'VALUE' 'INTEGER' );
};

'COMMENT'
***** Data Declarations *****
'INTEGER' 'ARRAY' SIDE ARRAY (0:2);
'BYTE' COUNTER;

'COMMENT'
***** Code *****

'COMMENT' Read in the three sides of the triangle;
PRINT TEXT ("Enter three lengths");
'FOR' COUNTER := 0 'STEP' 1 'UNTIL' SIDES PER TRIANGLE 'DO'
GET INTEGER ('LOCATION' (SIDE ARRAY [COUNTER]));

'COMMENT' Print the three sides;
PRINT TEXT ("Enter three lengths");
'FOR' COUNTER := 0 'STEP' 1 'UNTIL' SIDES PER TRIANGLE 'DO'
PRINT INTEGER (5, SIDE ARRAY [COUNTER]);

'COMMENT' Is it a triangle? ;
'IF' SIDE ARRAY [1] + SIDE ARRAY [2] < SIDE ARRAY [0] 'OR'
SIDE ARRAY [0] + SIDE ARRAY [2] < SIDE ARRAY [1] 'OR'
SIDE ARRAY [0] + SIDE ARRAY [1] < SIDE ARRAY [2]
'THEN' PRINT TEXT ("Not a triangle")
'ELSE' 'IF' SIDE ARRAY [0] = SIDE ARRAY [1] 'AND'
SIDE ARRAY [0] = SIDE ARRAY [2]
'THEN' PRINT TEXT ("Equilateral triangle")
'ELSE' 'IF' SIDE ARRAY [0] = SIDE ARRAY [1] 'OR'
SIDE ARRAY [0] = SIDE ARRAY [2] 'OR'
SIDE ARRAY [1] = SIDE ARRAY [2]
'THEN' PRINT TEXT ("Isosceles triangle")
'ELSE' PRINT TEXT ("Scalene triangle");

'END'
```

Figure 3 - The Triangle Problem

Asynchronous Point-to-Point Communications or Telephone-polling Multiple Computers

XOREN COMMS SOFTWARE- Software with ten years' pedigree

Throughout the 1980s, Xoren Computing pioneered the development of inexpensive yet reliable communications software. Today, the XOREN IPL-11 range is renowned for enabling automated file transfer between any quantity and combination of the major minis, micros and PCs in configurations which will become more and more commonplace through the 1990s.

Leading with Automated Connectivity

Although many other communications packages can asynchronously link PCs together, very few besides IPL-11 will orchestrate *fully automated connectivity* between multiple PCs and hosts.

Automated operation with IPL-11 gives users the freedom to perform unattended data communications 24 hours a day. Computers with modems operating up to 9600 baud can be scheduled to dial up remote computers over ordinary telephone lines day or night to send and receive files. Any IPL-11 version can link and transfer files to any other IPL-11 version, regardless of the computers' operating system. Once the files have arrived safely, telephone lines are disengaged.

Leading with Customer Support

Key to the success of the IPL-11 range is an emphasis by Xoren on customer support.

Because communications is notorious for being a difficult application to implement, Xoren offers installation and support services to ensure the software can operate according to the *user's* requirements.

This principle has made IPL-11 communications software the obvious choice for UK and European companies. Because IPL-11 is a British product, developed and supported in the UK, Xoren is strategically placed to support customers when they need support most - during regular business hours.

Call or write to Xoren today - the most practical solution to automated data communications has always been right on your doorstep.

XOREN IPL-11 Developments over a Decade

1980: Xoren launches IPL-11.

Features include peer-to-peer file transfer under operator control, CRC error-checking on individual packets and queuing of file transfer commands.

Original versions introduced for RSX-11M and VMS for DEC PDP-11 and VAX.

1981: Remote activation facility incorporated allowing file transfers when remote computers are unmanned.

RSTS/E, RT-11 and RSX-11M PLUS versions released for DEC PDP-11.

1982: Terminal emulation facility introduced enabling the use of a terminal on a local computer as a terminal on a remote computer thereby allowing control of file transfer sessions from a single terminal.

TSX PLUS version released for DEC PDP-11.

1983: Option to control file transfers from command files as an alternative to control from operator's keyboard.

P/OS version released for DEC Professional.

1984:

Mechanism built into the package protecting against "message bouncing" due to line noise when computers remain connected and the package is not in use.

MicroRSX and MicroRSTS versions released for DEC MicroPDP-11.

1985: Commenced development of new portable versions written in the programme language 'C'.

1986: First releases of new portable versions written in 'C' for PC-DOS, MS-DOS, UNIX, AIX and VMS

1988: PC versions enhanced with improved terminal emulation including VT100 emulation, keyboard mapping and facilities to define function keys.

1987: Portable versions support simultaneous multiple links IPL-11 wins ICP Million Dollar Award.

1989: Comprehensive upgrade for unattended operation of multiple PC/host links supporting auto-dialling modems.

1990: Release of MULTI-POLL range of polling software incorporating IPL-11 allowing PCs or other computers at multiple sites to be telephone-pollled by a central host computer system to transfer data to and from the central system.



XOREN COMPUTING LTD
28 MADDOX STREET
LONDON W1R 9PF
TELEPHONE 071-629 5932
FAX 071-629 5432

CIRCLE NO. 093

I am interested in Xoren Comms Software:

Name _____

Position _____

Company _____

Address _____

Postcode _____

EXE 7/90

Return to: Xoren Computing, Freepost 27,
London W1E 5JY.

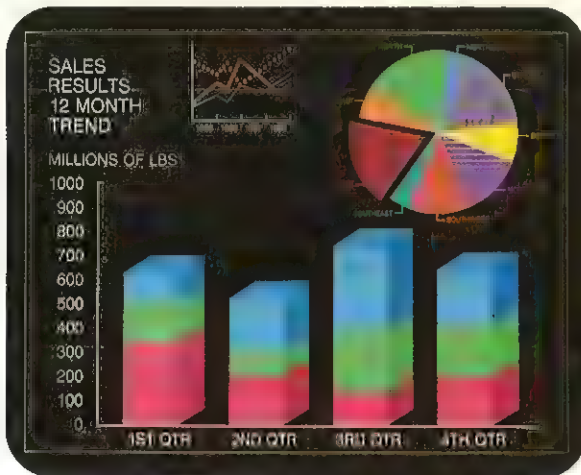
dGE

VERSION 4

GRAPHICS FOR dBASE

BITS PER SECOND

BITS PER SECOND LIMITED
14 REGENT HILL BRIGHTON BN1 3ED
TELEPHONE: 0273 727119
FAX: 0273 731925



dGE extends the database language of your choice to give full colour graphs and charts from **within** your programs.

dGE version 3 was voted the "best graphics tool for the database manager" for the second year running by readers of the influential U.S. magazine DataBased Advisor.

Now Version 4 is here with even more features:

- SuperText with variable sizes and face styles
- SuperIcons loaded from file
- SuperText and SuperIcon editor that can import PCX images
- Built-in Vector Text functions
- Gantt charts
- XOR capability added to many functions
- New interfaces to Clipper 5, FORCE and Vulcan
- Mouse "hot" regions
- A Norton Guide to functions

dGX, the Graphics Design Centre, has been completely revised, adding:

- Sophisticated database query and summary
- Dynamic scaling
- Import of PCX images
- Statistics functions
- Gantt charts, hi-lo and 3-D bars

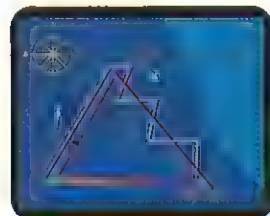
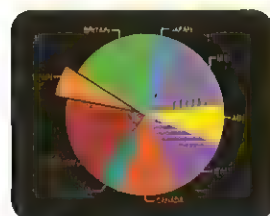
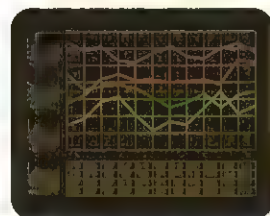
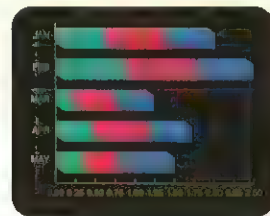
Version 4 comes complete with all the interfaces to dBASE and its dialects, the 'C' language, RBASE compiler and Vulcan

dGE supports CGA, EGA, VGA and Hercules modes and will output images to matrix printers, lasers and plotter

Call for information

Price £245 + VAT direct from Bits Per Second Ltd or from your authorised dealer.

dGE IS A TRADEMARK OF BITS PER SECOND LTD
 OTHER BRAND AND PRODUCT NAMES ARE TRADEMARKS OF THEIR RESPECTIVE HOLDERS



CIRCLE NO. 113

PROLOG + OBJECTS =

- Fast incremental compiler
- Rich development environment
- Editor, debugger, x-ref etc.
- Windows, graphics, icons

- Object-orientated programming
- Classes and instances
- Methods and super-methods
- Multiple inheritance



QUINTEC OBJECTS

- Object-oriented syntax for Prolog
- Rich windows-based development environment
- Graphical objects browsing/development tools

- Fully integrated with the underlying Prolog
- Fast incremental compilation
- From workstations (DECwindows) to PC (MS-Windows)

VAX, DECstation (VMS/ULTRIX, DECwindows)
 PC/386 (DOS, MS-Windows, SUN (UNIX))

CIRCLE NO. 088



**QUINTEC
 SYSTEMS
 LIMITED**

**MIDLAND HOUSE
 WESTWAY
 BOTLEY
 OXFORD OX2 0PL
 (0865) 791 565**

A feature which fits together well with code inserts is that of the Macro. The language defines the features and use of a macro preprocessor, whose function is to replace one section of source code with another - at compilation time. The informal syntax of the macro-definition is:

```
'DEFINE' NameOfMacro
      < BodyOfMacro >
```

where BodyOfMacro is a text string enclosed in double quote marks. Macro definitions can be used to remove detailed operations from the source code. This could be used with the previous example as follows:

```
'DEFINE' Save
" 'CODE'
  'BEGIN'
    Assembler Statements...
  'END'; "
```

Within the actual source code, only the statement SAVE would be used. Macros are also useful alternatives to procedures, to gain execution time, at the expense of code size.

The Triangle Problem

From what has gone so far, the source code for the Triangle Problem (Figure 3) should be self-explanatory. However, one point is worth a mention. The language has few facilities for the handling of string I/O operations. These are usually written by the user, as and when required. In this particular example, the I/O routines (PRTTXX, GETINT and PRTINT) are declared to be outside the program. To improve the readability of the code, these names have been redefined using macros.

CORAL 66 was the unmatched king of real time programming for many years. Why didn't it become widely used when the microprocessor came along? The answer is simple: cost. CORAL 66 compilers have always been comparatively expensive, compared to other languages. Second, the development environments have also been expensive when compared with, for instance, PC-based systems.

In 1980, if you wanted to develop software for, say, an Intel 8085, you would normally need an Intel MDS and a compiler costing

about £2,500. When the 8086 compiler was released (again hosted on the Intel MDS) the price was a staggering £6500. Later, a US company produced a software package which made a standard PC look like an MDS - and compiler costs fell sharply. But, by then, newer languages (such as real time Pascal) had arrived on the scene - and that was the beginning of the end for CORAL 66.

EXE

Dr Jim Cooling is a senior lecturer in the Department of Electronic and Electrical Engineering, Loughborough University of Technology. He has been deeply involved in the design, development and production of real-time systems for many years, and has published regularly on the subject.

Many thanks to Keith Rotton of Transmittion Ltd, Ashby De La Zouch, for producing and testing the triangle problem code.

If you are interested in reading further on this subject, then Dr Cooling recommends CORAL 66 Programming (author J T Webb, published by NCC Publications) and his own Software Design for Real-time Systems (published by Chapman and Hall).

The Complete Dongle Range for all your Software Protection Needs



- ★ Serial, Parallel or Internal Bus
- ★ Random number generator type
- ★ Deal directly with the people who created the product.
- ★ Great value at around £24



Further information direct from:

Data Encryption Systems Ltd
Edbrook House, Cannington,
Bridgewater, Somerset. TA5 2QE
Fax: (0278) 653300
Telephone: (0278) 653456



A DOS Debugger

WatchWord is a debugger by Robert Schifreen that watches a word in the PC's memory, and prints its contents whenever they alter.

When I'm poking around inside DOS, or trying to find why my program's not working properly, I frequently resort to the DEBUG debugger, because it's small and simple. However, one thing that DEBUG doesn't do is to watch the status of a piece of memory, and tell me whenever that area of memory changes. It's not as if you need a 386 machine to do this, either. WatchWord is designed to show just how easy it is to write a simple debugger that allows you to watch memory.

WatchWord is a TSR that sits on two different PC interrupts. First, it sits on the timer interrupt, 1Ch, and this part of the program gets called 18.2 times a second by the PC hardware. Every time this routine is called, it looks at the contents of a specified word in memory. If the contents have changed since the last time the program looked, the new contents (as a word, not a byte) are printed to LPT1. If the contents are the same, no action is taken.

The other interrupt that WatchWord traps is Int 15h, and this provides the interface between your program and the WatchWord debugger. To tell WatchWord which word to watch for, execute an Int 15h call, with AX set to 0FFFFh, and the address to watch

in ES:DI. So, if you want to watch the DOS clock change every second, execute Int 15h with AX = 0FFFFh, ES = 0 and DI = 046Ch, because the clock tick word is at 0:46Ch.

As the program stands, the watched word defaults to the keyboard status byte at 0:417h. This byte is a bit-mapped area that contains the current state of the Ctrl, Alt and Shift keys. So, once you've assembled WW.COM, and installed it by typing WW at the DOS prompt, press the Ctrl key and hold it down. Your parallel printer should print a value. Now, release the key and a different value will be printed.

The best use for WatchWord is for debugging your own programs. For example, if you want to find out when a certain routine is getting called, have that routine keep a byte in memory, and have that byte set to one every time the routine is entered, and to zero when it is left. Now, if you point WatchWord at that byte, you'll be able to monitor it.

There are a number of ways in which WatchWord can be enhanced. It was, after all, written in a couple of hours, purely to demonstrate that the basics of a debugger can be created quite easily. Feel free to add

a switch to turn printer output on and off (I suggest using Scroll Lock), and the ability to watch a whole block of memory rather than a single word.

The major problem with WatchWord is its timing resolution. The PC automatically generates an Int 1Ch every 18th of a second or so, and I have used this facility to allow WatchWord to watch memory. There is a well-documented method (found, for example, in *The new Peter Norton Programmer's Guide to the IBM PC and PS/2*) of reprogramming the PC's timer chip to increase the number of times per second that an interrupt is generated, and this could easily be used to increase the resolution by a factor of, say, four. If you do this, though, you will have to get involved with intercepting the Int 08h handler. This is the handler which actually receives the hardware interrupt from the clock - Int 1Ch is called from it. If the original Int 08h handler is called at a higher frequency, there are some bizarre side effects. The Norton Utilities SI index of your machine's performance will go down. The floppy disk motor will cut out before the read/write head has reached the selected track. The MS-DOS clock/calendar will run fast, and you'll find that your days start to fly by!

EXE

```

; WW.ASM - Watch a Word
; -----
code segment
assume cs:code,ds:code
org 0100h

start:
jmp init

intenc1c:                ; Come here 18.2 times per second

jmp watch

watchword_seg dw 0        ; Segment of word to watch
watchword_off dw 0417h    ; Offset of word to watch
prev_cont     dw 0        ; Previous contents
prtstr        db 'XXXX:XXXX' = XXXX',0Dh,0Ah,0

watch:
push si

push es
push di
push ax
push bx
push dx

mov es,cs:watchword_seg
mov di,cs:watchword_off
ax,word ptr es:[di] ; get contents of watched word
mov bx,word ptr cs:prev_cont
cmp ax,bx           ; has the word's contents changed?
je no_change

mov word ptr cs:prev_cont,ax ; The word has changed so print it
mov bx,ax
mov si,offset cs:prtstr ; Construct the print string
inc si                  ; point si to seg addr in print str
mov ax,word ptr cs:watchword_seg
call bin2asc
mov si,offset cs:prtstr
add si,6
mov ax,word ptr cs:watchword_off
call bin2asc

```

(continued)

Figure 1 - Listing of WW.ASM

CALL FOR OUR NEW CATALOGUE

ASSEMBLERS		Snoobi 4+		\$60.00		GSS Graphics Dev. Toolkit		\$389.00		Periscope II		\$110.00		PC Fortran Tools		\$99.00	
386 ASM Link & Locate		\$435.00		Trilogy		\$60.00		GSS Kernel		\$299.00		Periscope II-X		PC Yacc		\$249.00	
MASM v.5.1		\$289.00		PASCAL				GSS Metafile Interpreter		\$179.00		Periscope III		PolyAWK		\$285.00	
OPTASM		\$279.00		Microsoft Pascal		\$189.00		HALO-88		\$245.00		Periscope I Rev 3		PolyAWK OS/2		\$115.00	
Turbo Assembler/Debugger		\$289.00		Oregon Pascal-2		\$149.00		HOOPS		\$360.00		Periscope IV (16 MHz)		QPASER Plus		\$180.00	
				ProPascal		\$235.00		HOOPS 32		\$349.00		Periscope IV (20 MHz)		Text Management Utilities		\$75.00	
BASIC				MetaWare Professional Pascal		\$560.00		MetaWINDOW/Plus		\$169.00		Periscope IV (25 MHz)		Turbo Professional		\$75.00	
Microsoft Professional BASIC		\$270.00		MetaWare Prof. Pascal 286		\$375.00		sCLASERplus		\$299.00		Plus Board with 512K					
BASIC Interpreter		\$215.00		Turbo Pascal v.5.5 with objects		\$289.00		SilverPaint		\$60.00		Plus Board with OK					
Quick Basic U.K.		\$60.00		Turbo Pascal Professional		\$159.00		SilverPak		\$179.00		Serial Test					
True Basic		\$65.00						Slate		\$215.00		Soft Probe II/TX					
Turbo Basic		\$65.00		PROLOG				Universal Graphics Library		\$130.00		Soft-X-lore					
				Arity Combination Pack		\$685.00		OP. SYSTEM ENHANCEMENTS				Sourcer + BIOS Pre-Processor		\$89.00			
				Expert System Dev. pack		\$185.00		C Sharp		\$300.00		DOCUMENTORS					
				Intelligence Compiler		\$299.00		DESQview API C Lib/Ref		\$125.00		ASM Flow		\$60.00			
				MicroPROLOG Professional		\$425.00		Multi C		\$89.00		Clear+ for dBASE		\$125.00			
				MProlog		\$299.00		Multi Dos Plus		\$299.00		Clear+ for C		\$125.00			
				Professional Prolog II Plus		\$299.00		OS/2 Programmer's Toolkit		\$215.00		dAction		\$255.00			
				Professional Prolog		\$545.00		OS/286		\$299.00		dAnalyst		\$139.00			
				Prolog Advanced Toolkit		\$90.00		OS/386		\$299.00		Design/2.0		\$221.50			
				Prolog Compiler		\$120.00		Resident C		\$120.00		dFLOW		\$89.00			
				Prolog Compiler/Interpreter		\$399.00		Sapiens V8		\$180.00		Documenter		\$179.00			
				Prolog-86 Plus		\$150.00		Synergy Toolkit		\$239.00		Flow Charting II Plus		\$139.00			
				SQL Development Pack		\$185.00		Time Slicer		\$179.00		FoxToolBox		\$179.00			
				Turbo Prolog		\$75.00		VMFM		\$115.00		PolyDoc		\$169.00			
				AI AND OBJECT ORIENTED				Wendin PC UNIX		\$80.00		EDITORS					
				C-Talk		\$95.00		Wendin PC VMS		\$60.00		BRIEF v.3.0 NEW		\$159.00			
				C+ Objects Vol 1 with Source		\$275.00		Windows Development Kit		\$295.00		BRIEF v.3.0 UPGRADE		\$260.00			
				CzPERT		\$245.00		QUERY				dBRIEF OS/2		\$159.00			
				COMMUNICATIONS				CBTREE/SQL		\$239.00		dBRIEF v3.0 NEW		\$279.00			
				AdComm		\$185.00		CQL Query System		\$239.00		BRIEF with dBRIEF		\$229.00			
				C ASYNCH MANAGER		\$110.00		dQUERY/Lib		\$179.00		C-EDIT		\$239.00			
				Communications Library		\$145.00		db RETRIEVE		\$239.00		EMACS		\$199.00			
				Comm Pak with Breakout		\$195.00		XQL		\$480.00		Epsilon		\$110.00			
				Multi Comm		\$89.00		Xtrieve PLUS		\$360.00		ME Editor with C Source		\$115.00			
				Net Lib		\$125.00		Network SQL		\$385.00		Micro ISPF		\$110.00			
				NET-TOOLS		\$95.00		REPORT				Norton Editor		\$49.00			
				SilverComm		\$120.00		CB/IQ Report		\$139.00		PC/EDIT Plus		\$79.00			
				Comm Plus		\$60.00		Coreports		\$179.00		Personal Rxxx		\$79.00			
				Turbo ASYNCH PLUS		\$80.00		Focus Report Writer		\$210.00		PI Editor		\$119.00			
				FILE MANAGEMENT				r-tree		\$179.00		SPF/PC		\$149.00			
				asmTREE		\$245.00		R&R Relational Reportwriter		\$95.00		Turbo Basic Editor Toolbox		\$259.00			
				B-Tree with Source		\$110.00		R&R Clipper Module		\$49.00		Vedit+		\$115.00			
				B-Tree for C		\$49.00		TEXT SCREENS				dBASE on-line		\$60.00			
				AXOS		\$245.00		Animator COBOL		\$120.00		Norton Guides for ASM		\$75.00			
				Btrieve v.5.01 Novell		\$155.00		Application Manager		\$450.00		Norton Guides for BASIC		\$75.00			
				Btrieve OS/2		\$370.00		C-Worthy v.1.1		\$160.00		Norton Guides for C		\$60.00			
				Btrieve Multitasking		\$370.00		C-Worthy + Forms		\$240.00		Norton Guides for OS/2		\$89.00			
				Btrieve/N v.5.01		\$370.00		C-Worthy + Forms + Source		\$395.00		Norton Guides for Pascal		\$75.00			
				Btrieve XENDX		\$370.00		C-Worthy All C compilers		\$480.00		Programmer's Library		\$239.00			
				C Index+		\$245.00		C-Worthy Forms only		\$80.00		LINKERS & LIBRARIANS					
				c-tree		\$245.00		C-Worthy Source code only		\$155.00		ALINK NEW		\$195.00			
				Code Base 4		\$185.00		C-Worthy OS/2		\$240.00		Archive Link		\$239.00			
				db FILE/RETRIEVE		\$375.00		C-Worthy + Forms for OS/2		\$320.00		Link & Locate ++		\$245.00			
				d-tree		\$395.00		C-Worthy + Forms + Source		\$315.00		LinkLok		\$245.00			
				DBC III		\$160.00		C-Worthy OS/2 & MS-DOS		\$795.00		PLINK86 Plus Now Polytron		\$495.00			
				dbc III Plus		\$320.00		C-Worthy + source		\$150.00		PolyLibrarian I		\$85.00			
				RealDL/I		\$599.00		Curses		\$75.00		PolyLibrarian II		\$125.00			
				WKS Library		\$125.00		Curses Window Dev. Kit		\$149.00		RTLink		\$180.00			
				GENERAL				Data & Windows		\$179.00		MAKE UTILITIES					
				ASMLIB		\$89.00		DataWindow		\$60.00		PolyMake v.3.1		\$125.00			
				C Tools Plus		\$79.00		dWindows for db3		\$285.00		MKS Make		\$99.00			
				C Utility		\$120.00		Face-it		\$179.00		MKS Make 5 User		\$320.00			
				C*Lib C Function Library		\$239.00		Flash-up w/Toolbox		\$289.00		MKS Make 10 User		\$515.00			
				Clipper toolbox		\$125.00		Forms II		\$179.00		PROFILERS & OPTIMISERS					
				DB/RA3		\$255.00		HI-SCREEN XL		\$290.00		Codesifter		\$275.00			
				dBase Program Utility		\$55.00		Multi Windows		\$299.00		DOSTRACE		\$259.00			
				dBase Program Utility Vol 2		\$55.00		On-Line Help		\$55.00		PC Metric		\$120.00			
				dBase Tools For C		\$80.00		PANEL PLUS		\$79.00		PROTOTYPERS & GENERATORS					
				Finally		\$125.00		PC Forms		\$245.00		Automated Programmer		\$605.00			
				Functions with source		\$90.00		Pop Screen		\$245.00		Cogen		\$275.00			
				Integrated Dev. Libraries		\$90.00		POWER SCREEN		\$110.00		COMPEDITOR		\$110.00			
				Pascal Toolkit		\$75.00		Pro Screen		\$120.00		Dan Bricklin Demo II		\$125.00			
				Pascal Tools 1		\$90.00		SCREENIO		\$120.00		Data Boss for C		\$245.00			
				QuickPak Professional		\$79.00		Screen Sculptor		\$230.00		Data Boss for Pascal		\$239.00			
				Quick Tools Library		\$90.00		ScreenStar		\$230.00		Instant Replay III		\$89.00			
				Spindrift Library		\$160.00		Screenplay		\$230.00		Matrix Layout v.2.0		\$129.00			
				Super Functions		\$75.00		SoftCode		\$230.00		Sycero db (U.K.)		\$595.00			
				Tom Rettig Help Combo		\$75.00		SoftCode Template		\$230.00		Sycero db Net (U.K.)		\$745.00			
				Tom Rettig Library		\$75.00		Softscreen/Help Source		\$230.00		SYNTAX ANALYSERS & LINTS					
				Turbo C TOOLS		\$79.00		Sourcewriter		\$230.00		PC Lint v.3.0		\$89.00			
				Turbo Plus		\$80.00		UI Programmer Dev. Edition		\$230.00		Pre-C		\$150.00			
				Turbo PowerTools Plus		\$79.00		V Screen		\$230.00							
				Turbo Functions		\$89.00		VCScreen		\$230.00							
				GRAPHICS				Vitamin C		\$230.00							
				dBase Tools for C Graphic		\$55.00		VLIB with Source		\$230.00							
				dGE v.3.0		\$125.00		Vermont Views		\$230.00							
				EGA Toolkit		\$149.00		Z View		\$230.00							
				Essential Graphics		\$180.00		DEBUGGERS & DISASSEMBLERS									
				Flipper		\$120.00		386 Debug		\$125.00		C Programmers Combination		\$79.00			
				FontWINDOW/Plus		\$170.00		Breakout-II		\$159.00		C ToolSet		\$59.00			
				GEOGRAPH Professional		\$239.00		D'Scope 2		\$159.00		MKS LEX & YACC		\$165.00			
				Graphicmatic CGA & EGA		\$135.00		dBug for dBase		\$125.00		MKS Toolkit v.3.1 NEW		\$149.00			
				GraphicC		\$245.00		DIS*DOC		\$79.00		MKS Toolkit v.3.1 UPGRADE		\$250.00			
				OTHER LANGUAGES				TEXT & UNIX TOOLS									
				ACTOR		\$310.00		C Programmers Combination		\$79.00							
				ACTOR Language Extension		\$60.00		C ToolSet		\$59.00							
				CCSM Mumps Single User		\$65.00		MKS LEX & YACC		\$165.00							
				PC Logo		\$90.00		MKS Toolkit v.3.1 NEW		\$149.00							
				RPG II Complete		\$995.00		MKS Toolkit v.3.1 UPGRADE		\$250.00							
				Smalltalk/V		\$65.00											
				Smalltalk/V 286		\$129.00											

Figure 1 - Listing of WW.ASM (continued)

```

mov si,offset cs:prtstr
add si,14
mov ax,bx
call bin2asc ; Print string is now formatted

; Now print string

mov si,offset cs:prtstr ;String to output
xor dx,dx ;Printer LPT1:
prloop:
mov ah,0 ;Print string
mov al,cs:[si.]
cmp al,0
je no_change
int 17h
inc si
jmp prloop

no_change:

pop dx
pop bx
pop ax
pop di
pop es
pop si

db 0EAh ; Far jump to original Int 1Ch handler
int1pc dw 0 ; Address of orig handler, filled by
intcs1c dw 0 ; installation code.

intent15:

cmp ax,0FFFFh
jne done
mov word ptr cs:watchword_seq,es
mov word ptr cs:watchword_off,di

done: db 0EAh ; Far jump to original Int 15h handler
intip15 dw 0 ; Address of orig handler, filled by
intcs15 dw 0 ; installation code.

bin2asc proc near ;Convert AX to ASCII and put at CS:SI
push ax
mov al,ah
call convai
pop ax
convai:
push ax
push cx
mov cl,4
ror al,cl

```

```

pop cx
and al,0Fh
add al,30h
cmp al,39h
jbe bin1
add al,7
bin1:
mov cs:[si],al
inc si
pop ax
and al,0Fh
add al,30h
cmp al,39h
jbe bin2
add al,7
bin2:
mov cs:[si],a.
inc si
ret
bin2asc endp

init: ; Install WW

mov ax,cs ; Set DS = CS
mov ds,ax

mov ah,35h ; Get old interrupt vector
mov al,1Ch
int 21h
mov word ptr cs:intcs1c,es ; Save in long JMP
mov word ptr cs:intip1c,bx
mov ah,25h ; Set new interrupt vector
mov al,1Ch
mov dx,offset cs:intent1c
int 21h

mov ah,35h ; Get old interrupt vector
mov al,15h
int 21h
mov word ptr cs:intcs15,es ; Save in long JMP
mov word ptr cs:intip15,bx
mov ah,25h ; Set new interrupt vector
mov al,15h
mov dx,offset cs:intent15
int 21h

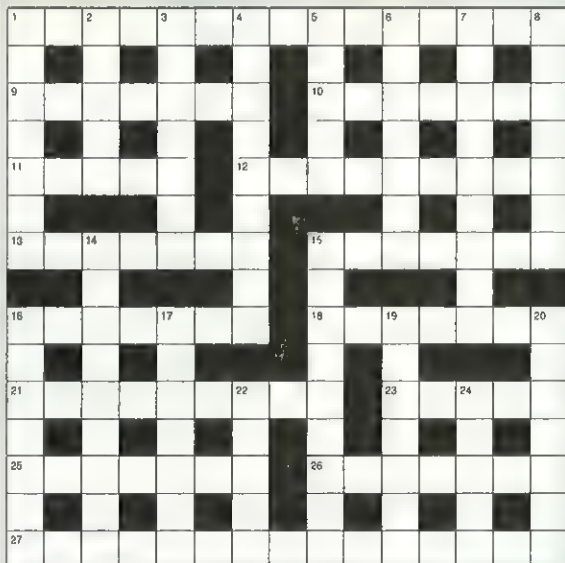
mov ax,3100h ; End, with errorlevel 0
mov dx,(init-start):16+1 ; Number of resident paragraphs
int 21h ; Done
code ends
end start

```

.EXEWORD

ACROSS

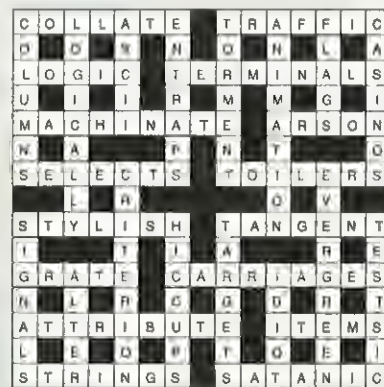
- 1 Getting into IT from scratch (15)
- 9 Trick cyclist for system? (7)
- 10 Null and void (7)
- 11 Bird houses set in structured code (5)
- 12 Climb step by step (9)
- 13 Links between structures (7)
- 15 Bars backward Gas buyer from water holes (7)
- 16 Signals controlled by signals (7)
- 18 Make a transposition error, say (7)
- 21 They oppose current change (9)



- 23 Peripheral to the sound sensor, for instance (5)
- 25 Real opposed to 17 not integer (7)
- 26 Momentum - laziness to others (7)
- 27 Apparently unnecessary test on parity, say (10,5)

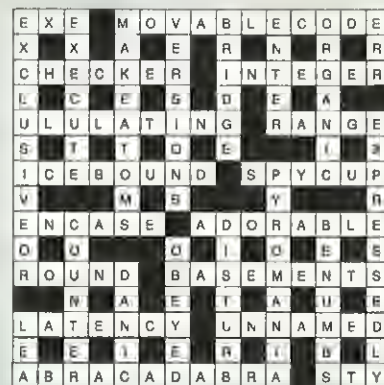
DOWN

- 1 Path for 16ac getting a tunnel (7)
- 2 Sounds of discontent (5)
- 3 Journeyman from Homer to Joyce (7)
- 4 N-type or P-type (9)
- 5 Capital charge on chip (5)
- 6 Suitability of fast train crossing loch... (7)
- 7 .. as I train rye to make a journey (9)
- 8 Neutralises gas on logic circuits (7)
- 14 Gave up sailor and individual note (9)
- 15 Re-starts work with a full instruction set (5,4)
- 16 Schmitt made one but not for his gun (7)
- 17 Font chip I broke up in a story... (7)
- 19 ... and the way oily acid took a chip (7)
- 20 Put down for yourself sound sensor on omr input (7)
- 22 Final character of old (5)
- 24 Program name and state reward (5)



.EXEWORD MAY

Our apologies, but due to a printing error, May's crossword solution did not appear last month.



.EXEWORD JUNE

Here's what they say about Zortech C++

"Zortech is a truly fine compiler...If you've been waiting for a major player to offer a professional C++ development system for OS/2 and Windows, as well as DOS, wait no longer... Zortech has it!"

Richard Hale Shaw, PC Magazine, p.38, March 13, 1990

"Zortech C++ is one of the best MS-DOS products I've had the luck to use.....I can highly recommend the Zortech 2.0 release."

Scott Robert Ladd, Dr. Dobbs Journal, pp. 64-73, January 1990

"Zortech has done a commendable job with C++ 2.0 and I recommend it highly...The debugger is impressive...Get the Developers version...it's worth the money."

Bruce Eckel, Micro Cornucopia, pp. 8-17, March 1990

"The Zortech compiler is a well-crafted product; in particular it is efficiently documented and it produces high quality object code. I can recommend it as a fine package"

Paul G Smith, .EXE Magazine, pp. 16-22, March 1990

AT&T™ C++ V2 Specification

- ✓ Multiple Inheritance
- ✓ Type Safe Linkage
- ✓ Pointers to Members

Compiler Features

- ✓ Native code compiler with separate global optimizer
- ✓ Improved MSC Source Level Compatibility
- ✓ MS Windows™ Compatible
- ✓ CodeView™ Compatible
- ✓ Fast Graphics Library with C++ interface
- ✓ Easy to use TSR functions
- ✓ Standard Library Source Code included with Developer's Edition
- ✓ Seamless LIM/EMS Support via new handle pointers or directly via EMS library functions.
- ✓ Full MS Mouse Library
- ✓ OS/2 Compiler Option
- ✓ 99% ANSI C Compatible
- ✓ Improved code size/speed

C++ Source Level Debugger

- ✓ Also Debugs C
- ✓ Assembler Debugging with access to registers and memory.
- ✓ 16 Debugging Windows
- ✓ Multiple Statement Lines
- ✓ Break/Trace/Watchpoints
- ✓ Dual Monitor Support
- ✓ Full C++ name unmangling for easy use
- ✓ Block memory write protect

C++ Tools Classes

- ✓ 25 C++ Classes with full source code
- ✓ Includes new Text User Interface Classes
- ✓ Event Queue, BCD Maths, Linked Lists, Money, DOS error handling classes, text windows and editing classes, virtual arrays, time and date handling, directories and filenames, interrupt vectors, etc...

PRICES

C++ Compiler £129.95
C++ Debugger £99.95
C++ Tools £99.95
Library Source £99.95
Save £130 - Get the Developer's Edition for only £299.95 (includes all the above items).
OS/2 Option £99.95
C++ Video £299.95

USA: Zortech Inc.

4-C Gill Street
WOBURN MA01801
Voice: 617-646-6703
Fax: 617-643-7969

EUROPE: Zortech Ltd.

106-108 Powis Street
LONDON SE18 6LU
Voice: 44+ 81-316-7777
Fax: 44+ 81-316-4138

CIRCLE NO. 097

"ANNOUNCING V2.1" 640K Memory Barrier Smashed!

- **New** VCM™ (Virtual Code Manager) technology
- **New** Rational DOS Extender technology for compiling/debugging massive programs
- **New** Virtual C++ Source Level Debugger requires only 4k RAM!
- **New** Remote Debugging via serial port
- **New** Powerful Environment with Browser
- **New** Completely Revised & Expanded C++ Tools
- **New** Improved Compiler Optimization

Zortech VCM™ for DOS

With Zortech's Virtual Code Manager (VCM) you can compile standard MS-DOS applications containing up to 4Mb of code. VCM is a sophisticated virtual memory system that dramatically improves performance over conventional overlay methods. Naturally, our debugger understands VCM too!

Rational™ DOS Extender Technology...

Version 2.1 incorporates this new technology for compiling and debugging really big programs on 286, 386 or 486 based PC's. You can also use V2.1 together with Rational Systems DOS Extender (purchased separately) to produce your own applications which can access memory beyond the 640k DOS limit.

C++ Debugger in 4k RAM!

Zortech's Virtual C++ Source Level Debugger can now locate itself in extended memory on 386 machines. This requires only 4K of conventional RAM!

STOP PRESS - NEWS FLASH

386 Compiler/Debugger Option (using Phar Lap DOS Extender), UNIX 386 Compiler and OS/2 Debugger all available soon. Also new C++ Classes and Addison Wesley ZTC++ book.

ORDER/UPGRADE HOTLINE 44+81-316-7777

Command arguments from program output

In the first of a new UNIX column, Peter Collinson highlights a feature of UNIX's handling of command line arguments that will be a surprise to MS-DOS programmers.

I felt that I would like to start my regular column with something that I think is really 'UNIX' - the back-quote operator. This is part of UNIX shell syntax which allows the output from one command or command pipeline to be directly used as arguments to a program. The topic also allows me to highlight some differences between the MS-DOS way of doing things and the UNIX way - just remember that one is never *better* than the other, simply different.

The UNIX command line consists of a command name followed by a number of arguments. For example:

```
ls -la /bin Makefile
```

is a command name `ls`, some program arguments `-la` and some filenames. Another familiar piece of syntax allows filename expansion:

```
ls -l *.c
```

lists all the files whose name ends in `.c`. Of course, there are other forms of expansion permitted by the shell; these were discussed in June's article on regular expressions.

It is important to realise that the expansion is done by the shell *before* the `ls` command is called; the string `*.c` is not passed into the command for it to interpret. The shell sees the star, reads the directory and performs the necessary pattern match while constructing an argument list. The expanded list is sorted and is finally passed into the program as a number of strings.

It is possible to be surprised by the order in which this all happens. For instance, when typing:

```
ls -l * > output
```

the output file will contain its own name. Worse, the size of the file will be shown as zero. This piece of counter-intuitive behaviour is easily explained when it is realised that the output file is created before the `ls` command is even run. Also, when `ls` sees the file, it has just been created by the shell and really is zero length. However, this whole effect can sometimes be embarrassing.

Concentrating the code for all the complicated pattern matching and directory reading in the shell makes sense, since it exists in one program and does not have to be installed in every program on the machine. The expansion mechanism works well when the desired files are all in one directory and their names have a regular structure. It can be deficient if a more complicated selection process is needed.

Complicated Selections

Consider how you might perform a non-trivial, but nearly repetitive, edit to a number of files; for example, adding a parameter to all occurrences of calls to a routine. The way to do this is to use `grep` to find the files which contain the routine name, so each file can be edited in turn to make the required change. There is a chance that a file will be missed; this is lessened if the editor is called by a command like:

```
vi command.c main.c odds.c
```

since `vi` has commands for stepping through the files one at a time. However, there would be much less chance of a mistake if `vi` could be called with arguments supplied from the output of the `grep` command.

The back-quote operator provides exactly this functionality. It is a general mechanism for constructing argument lists from the output of a command or command pipeline. When a command is placed in back quotes in an argument list, it is run and its standard output is caught by the shell. The output is parsed and supplied to the command at the start of the line as program arguments. The example becomes:

```
vi `grep -l routine_name *.c`
```

`grep` is passed the `-l` option to suppress its normal output; instead it prints a list of filenames on its standard output. A filename will be included in the list when it contains the match string and it is guaranteed to only appear once. The shell reads the list and arranges to call `vi` with the derived arguments. `vi` will run with no arguments if the list is empty.

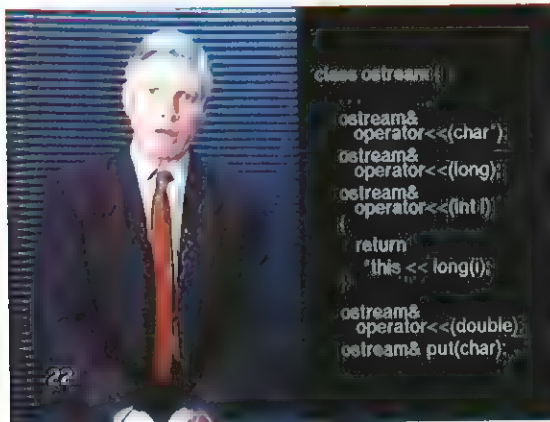
The back-quote operator simply turns a list of files coming from a program into a set of arguments. It considerably enhances the power of the command line, especially given that UNIX has many programs designed to generate a list of file names. Here is another example:

```
tar cf /dev/rst8 \
  `find . -mtime -5 -print`
```

This `tar` command copies files to cassette tape. The file names are taken from the standard output of the `find` command, which, starting from the current directory, will search for files which have been modified in the last five days. The `-print` option to `find` will cause a matched name to be listed on standard output. The shell uses this list as a set of arguments to the `tar` command and the appropriate files are written onto the tape. This command can form the basis of a file backup system.

VIDEO COURSE

Learn C++



NO RUSH!
NO TRAVEL!
NO HOTELS!
ONLY £299.95

Learn C++ Now!

The great exodus of programmers from C to C++ has begun! Since C++ builds on C, it's the easiest OOP language to learn. That's why it's called "the language of the 90's".

Why the rush?

PRODUCTIVITY! Yes, C++ programmers can write programs in less time requiring less maintenance. Large projects become much easier to manage.

Unfortunately, learning C++ can be very costly. Classroom instruction is expensive even without the travel and hotel costs. Of course, not learning will cost you even more in the long run. Now there is an alternative!

The top C++ video tutorial at the lowest possible price.

The C++ video tutorial from Zortech is the ultimate C++ training tool for work or home at only £299.95. It comes on six VHS video tapes containing 32 clear, extensive tutorials.

Used in conjunction with the concise workbook and tutorial disk, you will find everything you need for fast-track C++ tuition.

The course is generic (i.e. compiler and hardware independent) and is available with or without the award winning Zortech C++ Compiler for MS-DOS and OS/2.

Start writing C++ code within a week.

As a C programmer, you will start producing C++ code within a week of concentrated use of this course. Alternatively, spend just an hour a day watching the video and working through the suggested exercises to learn C++ in only six weeks!

The Leader in video tutorials.

If you don't already know C, you can join tens of thousands of programmers world-wide who have learnt C with the Zortech Complete C Video Course, described as:

"An excellent bargain ...
 ... I heartily recommend"
 Gary Ray, PC WEEK

Save your company thousands of £££'s.

One programmer can train for only £299.95, but you can train ten programmers for just:

C++ Course	£299.95
9 Extra Workbooks	£179.55
Total 10 students	£479.50

Yes! Only £47.95 each!
 (With all the FREE refresher courses you need!)

Now, Zortech with its new C++ Video Tutorial has refined the art of video tuition and presentation even further for the 1990's.

IN EUROPE CALL:

44- 1-316-7777

Only £299.95 complete

- Six Videos with 32 lessons
- 256 page workbook
- Tutorial disk
- Compiler & hardware independent
- NTSC or PAL format
- Tax deductible

Don't delay, order now!

Just mail the coupon or call the order hotline for same day shipment.

USA: Zortech Inc.,
 1165 Massachusetts Avenue,
 ARLINGTON, MA02174
 Voice: 617-646-6703
 Fax: 617-643-7969

EUROPE: Zortech Ltd.,
 106-108 Powis Street, LONDON
 SE18 6LU
 Voice: 44- 1-316-7777
 Fax: 44- 1-316-4138

ORDER FORM

Please rush me these items:

Qty	Description	Price
—	C++ Video Tutorial	£299.95
—	Extra C++ Workbooks	£. 19.95
—	Zortech C++ Compiler	£129.95
—	Developer's Edition	£299.95

EXE 3/90

Please add £5 + VAT postage in UK
 All UK orders please add 15% VAT
 International shipping charged at cost

Name _____
 Company _____
 Address _____
 Phone _____
 MC, VISA or CHQ _____ Expiry _____
 Card No. _____

CIRCLE NO. 098

Clip off
this coupon,
staple a business card
to it then return it to us
for a free
MicroMuse Information Pack

UNIX V 3.2

Complete Development Systems

For less than £3,000 MicroMuse can supply a fully functional 2 user development system complete with everything required to be up and running on the day of delivery!

Micro/lx 200	£2,995	Micro/lx 350	£3,995
--------------	--------	--------------	--------

20 MHz 80386 0 wait state system board
Socket for 80387 or Weitek
4 MB RAM expandable to 16
75 MB 18mS cached HD
1.2 MB floppy drive
2 serial, 1 parallel
101 keyboard
Mouse
VGA with 14 in paper white screen
Mini tower with 200W PSU

33 MHz 80386 0 wait state system board
Socket for 80387
4 MB RAM expandable to 16
75 MB 18 mS cached HD
1.2 MB floppy drive
2 serial, 1 parallel
101 keyboard
Mouse
1024×768×16 display controller
14 in multiscan screen (up to 1024×768×256)
Mini tower with 200W PSU

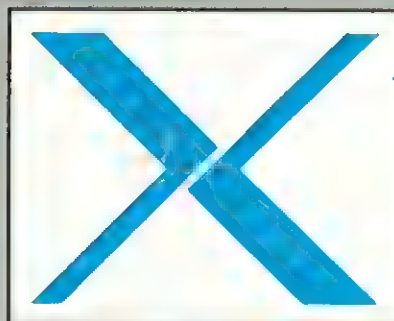
Systems also include Documentation, 1 year warranty
and complete Unix V 3.2 development environment

Options and Substitutions

Replace 75MB with 210 MB
Hard Disk £692
150 MB tape drive £686
Additional memory (per Megabyte) £80
8 line intelligent serial controller £794

Motif compliant graphical user
interface £395
Sun Microsystems PENFS
(includes TCP/IP) £350
Western Digital thinwire ethernet card £195
O'Reilly X windows manuals (3 vol set) £65

MicroMuse systems include:
SVID conforming Unix V 3.2 2 user licence
Standard AT&T C compiler
X windows release II V3, 02 development system
RFS (VPix & NFS optional)
Xenix file compatibility
TCP/IP
Xtoolkit and Xlib
Streams



486 systems and Unix networks also available

Micromuse systems are supplied with software installed and all components guaranteed to function properly with Unix.

If you are looking for an up to date, compatible development environment save yourself time, money and headaches!

CALL MICROMUSE NOW

MicroMuse Ltd., Unit 1L, 1st Floor, Chelsea Reach,, 79-89 Lots Rd., London SW10 0RN
Phone: 071-352 7774, Fax: 071 351 7834

MicroMuse

All Trademarks acknowledged

Thinking with Micros



UNIX

There are many situations where several operations are to be done on a set of unrelated files. Rather than typing the file names in again and again, they can be placed in a file, say `sourcedata`. The input to the file can be diverse, minimally using the combination of `ls` and an editor. Then using the syntax like:

```
cp `cat sourcedata` destination
```

a number of files will be copied.

First, the `cat` command runs, simply copying the list of files to its standard output. The shell captures the list and creates a set of arguments for `cp` from it. Finally, `cp` is called copying the set of files to the destination.

Interactive Selection

It can be tedious to create the source argument file, so the back-quote technique using `cat` is only appealing when the source file can be used more than once. For one off use, a tiny shell script called `pick` can be combined with the back-quote operator to allow all commands to have their arguments selected interactively. The job of the `pick` shell script is simple. It presents its arguments one by one to the user and prints a question mark. If the user answers `y`, the argument is copied to standard output, otherwise the argument is discarded. The simplest form of `pick` is:

```
#!/bin/sh
# Very simple pick command
for i in "$@"
do
    echo -n "$i? " > /dev/tty
    read answer
    case "$answer" in
        y*(Y*)) echo $i ;;
        esac
```

```
done < /dev/tty
```

So if we were copying files from the current directory, the `cp` command would look something like:

```
cp `pick *` destination
```

and if the directory contains `apple`, `grape`, `lemon` and `mango`, the `pick` dialogue would go something like:

```
apple? y
grape?
lemon? y
mango?
meaning that pick will print apple and lemon, and only those files will be copied.
```

`Pick` extends to all commands the flexibility offered by the `-i` option to `rm`. It is a good example of what happens so many times in UNIX:- simple operations combine to make some greater whole - and that's the UNIX philosophy.

EXE

Peter Collinson is a freelance consultant specialising in UNIX. In 1976, as a lecturer at the University of Kent, he was one of the first in the UK to become a UNIX system administrator/C programmer/kernel hacker/guru. He is now involved with consultancy, programming, training and yes... writing. He can be reached electronically as pc@hillside.co.uk (although your mailer might be happier to put the address the other way round) or by phone on 0227 761824.

Thanks to Brian Kernighan from AT&T Bell Labs, for permission to use `pick` in this article. `pick` is taken from the excellent book *The UNIX Programming Environment* (by Brian W Kernighan and Rob Pike, published by Prentice Hall, ISBN: 0-13-937699-2). The example shown here is rudimentary, and people intending to use `pick` seriously are strongly recommended to consult the book.

System Science

C & C++ COMPILERS

Microsoft C 6.0	£Call
Quick C with Quick Asm	£98.00
TopSpeed C ext	£285.00
Turbo C Professional	£155.00
Watcom C	£255.00
Watcom C/386	£645.00
Zortech C++ dev ed	£275.00

CROSS DEVELOPMENT

Introl C/Mod-2 68000	£1445.00
Aztec C Cross	£265.00
Avocet AvCase 8051	£705.00
IAR C Cross 8051 etc	£1095.00
2500AD Cross Assemblers	£135.00
IAR Cross Assemblers	£185.00
Avocet AvMac Cross Asm	£295.00
Simulators & others	£call

C GRAPHICS

Halo	£225.00
GSS Graphics Dev T'kit	£385.00
GSS XVT Windows	£385.00
Metawindows	£175.00
MS Windows SDK	£315.00
Essential Graphics	£275.00
PCX Programmers Toolkit	£165.00

FORTRAN

Lahey Fortran F77L	£395.00
Lahey F77L-EM/32 (386)	£795.00
MS Fortran-75 v.5	£245.00
RM Fortran Forte	£465.00
Salford FTN77/386	£695.00
ProFortran 77 (DOS&OS/2)	£295.00
InGraf for Fortran (source)	£175.00
many Fortran Libs	£call

C DATAFILE

Ctree source	£245.00
Btrieve Network	£395.00
XQL	£525.00
CodeBase 4	£195.00
db_VISTA III	£545.00

C SCREENS + COMMS

C Scape, Look & Feel	£315.00
Vermont Views (new ver)	£325.00
Panel Plus II	£275.00
Greenleaf Datawindows	£195.00
Aspen Curses (DOS & OS/2)	£115.00
Aspen Formation	£225.00
C Worthy	£call
Blaise C Asynch Manager	£115.00
Greenleaf Comms	£175.00
Essential Comms	£195.00

PASCAL + BASIC

Turbo Pascal	£75.00
MS Quick Pascal	£65.00
Object Professional	£95.00
B-Tree filer (M/U)	£120.00
Blaise Power Tools Plus	£89.00
Blaise Asynch Manager	£89.00
MS Basic Prof Dev System	£275.00
MS Quick Basic	£65.00
Ingraf Basic	£175.00
QuickComm	£95.00
Crescent QuickPak Prof	£129.00
Hammerley ProBas	£135.00

OTHER LANGUAGES

Smalltalk V	£69.00
TopSpeed Modula-2	£59.00
MuLisp-87 Interp	£175.00
Microsoft Cobol/2	£445.00
PC Logo	£75.00
UR/Forth by LMI	£225.00

PROGRAMMERS TOOLS

MMC AD Prog T'box	£165.00
Dan Bricklin Demo Roman II	£175.00
MKS Toolkit DOS	£155.00
VsDesigner (case)	£375.00
Brief Editor	£149.00
Epsilon Editor (DOS/OS2)	£145.00
Vq2 Editor (DOS & OS/2)	£125.00
Kedit Editor	£115.00
PVCS Prof	£375.00
PolyMake	£110.00

ASSEMBLERS & LINKERS

MS Asm 5.1 & Codeview	£95.00
Advantage Disasm	£145.00
Turbo Assembler/Debugger	£99.00
RTLink Plus	£345.00
Plink 86+ LTO	£525.00
Source & Bios (DisAsm)	£125.00

MEMORY & DISK + MATHS

386 Max Prof	£90.00
QEMM/386 5.0	£60.00
GRAM with Manifest	£55.00
Above Disc (LIM Emul)	£69.00
Norton Utilities Advanced	£79.00
Norton Backup	£89.00
SpinRite II	£69.00
VFeature Deluxe	£85.00
Vcache	£49.00
Derive Symbolic Algebra	£115.00
MathCad	£345.00
Microstat II	£295.00
Statgraphics	£545.00
Mathematica	£call

**A SAMPLE OF OUR EXTENSIVE RANGE
PLEASE ENQUIRE FOR UNLISTED ITEMS**

3-5 Cynthia Street,
London N1 9LF
Fax: 071-837 6411



Prices are subject to change and alteration Prices include UK Delivery

071-833 1022

CIRCLE NO. 100

Books

There's a distinctly networky feeling to this month's selection of books; with one volume devoted to Novell's Netware and the other to NetBIOS.

Effort Rewarded

NetBIOS is an IBM-originated 'standard' for peer to peer communications on PC-based local area networks. It comprises an API that programmers can make use of when they write their programs, and a set of protocols that allow packets of data to be sent and received within a higher level framework. This framework allows the management of complete communications sessions. There are alternatives to NetBIOS, such as Novell's IPX/SPX suite of protocols, but none are as pervasive as NetBIOS, which is available for a multitude of operating systems and networks.

If you need to learn about NetBIOS programming, *The C Programmers' Guide to NetBIOS* is a good place to start, but don't expect it to contain a gentle introduction or an 'executive summary'. The author of this book believes in going into detail, which he does as quickly as he can.

The detailed treatment of the subject matter is admirable and loads of sample source code is provided. The book includes an overview of how to issue NetBIOS commands, code to establish the presence of NetBIOS, explanations of NetBIOS command parameters, and explanations and sample code for all aspects of NetBIOS programming from name support through datagram management to session programming. The book even give sample source code and explanations of how to write a network conferencing application (not quite on the scale of CIX, but it is interesting nonetheless) and how to write device driver redirectors. The last two chapters provide an extremely useful reference guide to NetBIOS. A tear-off card is provided to request the sample code in disk form.

This is a book written by someone who knows his subject inside out. This will not always be true of its readers: the author does not provide a gentle introduction to his subject, and occasionally he goes into unnecessary detail. A considerable amount of space is given to a treatise on CRC programming (maybe Schwaderer's college thesis was about CRCs?). Not enough detail (ok, no detail whatsoever) is given to non-IBM implementations of NetBIOS. To read the book one might be forgiven, at times, for thinking that the IBM PC LAN was the only sort of network.

When I picked up the book the first time, I was expecting to be able to dive in and read it in fragments, extracting information as I required it. This is not possible, and one must read this book end to end for it to be useful (a long session reading it in the bath is to be recommended, from experience). Given the effort, the book pays dividends to its reader. On a recent project with which I have been associated, the *IBM NetBIOS Application Development Guide* (the official IBM reference to NetBIOS) lies hardly used, whereas Schwaderer's book is referred to constantly. If you need to learn about NetBIOS programming, buy this book.

Paul G Smith.

Title: *C Programmer's Guide to NetBIOS*

Author: W David Schwaderer

Publisher: Howard W Sams & Company

Price: £20.95 ISBN: 0-672-22638-3 Pages: 320



Weighty But Light

Novell Netware - The Complete Reference is another offering from the usually excellent Osborne McGraw-Hill stable. You will not, therefore, be surprised to learn that it is 9 inches high, 7.5 inches wide and 2 inches thick with a paperback cover. It also costs four times more than you would expect to pay for any non-computer paperback of similar size. I know that the Computer World is obsessed by standards, but this is no justification for the large number pages required to cover a given subject.

I could not find any reference to the background of the author, Tom Sheldon, other than the information on the back cover. This informs us that he is 'a fully authorised Gold Level Novell service and support technician. He manages the Technical Products Division for a chain of west coast Computerland stores and has worked extensively with network systems, including Novell.' Based on the information packed into the book, Sheldon seems to know his stuff. It could equally be that he is a very good précis writer, who has succeeded in condensing the seven or eight weighty manuals that come with Novell down to a less intimidating single volume. Whatever the truth of the matter, the result is a genuine high-level reference manual for Netware. This enables the user to find the answer to questions without the need to wade through the well-written, but very detailed, red books that come free with Netware.

The book begins with general sections on concepts and components of a network. I found these sections to be fairly weak, as Sheldon seemed both uncertain of what he wanted to say and how to say it. The result is a hotchpotch of views and ideas, that leave the reader with little benefit for making the effort to read them.

The standard of writing improves significantly as Sheldon moves on to cover the world of Netware, the products available and the features provided by each. A section on Novell's connectivity offerings is so good that they would do well to consider using it for their own product summaries.

Sections quickly follow covering planning and installing networks, creating directories, setting up security and day to day operations. This includes a comprehensive look at using all the various facilities and commands that come with Netware. If you want a quick summary that gives you the feel of a command or function, rather than an in-depth explanation, then you will almost certainly find what you're looking for here.

In summary, this is a well-written book that achieves its objective of providing a comprehensive coverage of the Novell range of products. It's not for the 'techie', but if you are new to Novell, or simply want to expand your general knowledge of the products, then 'The Complete Reference' will make a useful addition to the big red collection. It will definitely provide an insight into the sheer breadth of the Novell world.

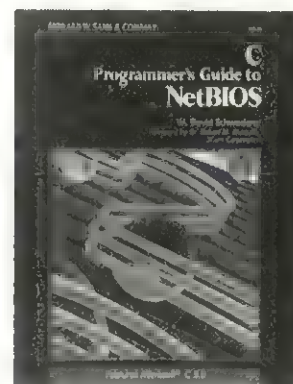
Dave Apsey.

Title: *Novell Netware - The Complete Reference*

Author: Tom Sheldon

Publisher: Osborne McGraw-Hill

Price: £35.95 ISBN: 0-07-881594-0 Pages: 746



EVERLOCK



COPY PROTECTION

Designed for clone compatibility & strength

- Thwarts ALL Bit-copy Software
- Protect any COM/EXE w/o Source changes
- Shut down Debug Tracing & Disassemblers
- Supports all Hard & Floppy Disk formats
- Restrict or allow Network Installs
- Limit program use by Date or # Executes
- Remotely reset Installs, Date or # Executes
- No damaged media or I/O plugs

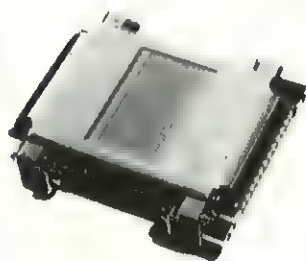
For IBM or Clones £465.00 + p&p £6.50
FREE INFORMATION & DEMONSTRATION
Limited Use Version £150.00 + p&p £6.50
(Prices exclude VAT)

USER
FRIENDLY
micro-computer
systems

ICS House, Edward Street
Redditch, Worcs B97 6HR
Tel: 0527 585550 Fax: 0527 65111

CIRCLE NO. 101

SOFTLoK™ & SOFTLoK PLUS™ software protection systems



- For IBM PC, PS/2 and compatibles
- Installed in seconds in the parallel printer port
- Assembler & high level source code provided
- Fast response
- Transparent to printer
- Small physical size

SOFTLoK

SOFTLoK units contain a code that can be read and verified by routines provided in assembler and a variety of high level languages.

Price: 10-99 units £12.00, 100+ units £8.00
Evaluation kit £20.00

SOFTLoK PLUS

SOFTLoK PLUS units contain a small area of non-volatile memory protected from unauthorised reading or writing by a password. Both the memory area and the password can be changed at any time by the routines provided in assembler & several high level languages. As the memory can even be altered in the field it can contain counters & expiry dates as well as serial numbers etc.

Price: 10-49 units £40.00, 50+ units £35.00
Evaluation kit £40.00

SOFTLoK International Limited
Tel: (0254) 772220 Fax: (0254) 705956

CIRCLE NO. 103

CTM Programmer's / MPW Toolbox / PC Sun

The **C Programmer's Toolbox**, by MMC AD Systems, enhances the tasks of documenting, testing and maintaining efficient C software. With over 20 tools, the **C Programmer's Toolbox** supports ANSI, Microsoft, Quick, Turbo, Sun OS, Mac MPW and Think C. The **Toolbox** handles any sized program by using virtual memory.

A sample of the tools included:

Cflow - function hierarchy and interdependence.

Clint - find those syntactically correct gotcha's.

CPrint - reformats and beautifies your C source.

CritPath - to help improve program performance

CXref - and check usage of symbols in C source.

PMon - monitor a programs execution ... more

C Programmers Toolbox - PC Pack £165.00

MPW £245.00

Sun OS £420.00

(Prices are exclusive of VAT)

System Science

3-5 Cynthia Street
London N1-9JF

Tel: 01-833 1022
Fax: 01-837 6411

CIRCLE NO. 102

BRENT

Communication and Telemetry Consultants

"MaxPro - the ultimate in hardware for software protection"



The MaxPro Data key fitted to the Parallel Port is totally transparent to the end user.

Software customers make as many backups as are required but of course only the machine using the individually coded MaxPro key will run the program.

BEWARE THE PIRATE'S PATCH

High security encryption is created by the Simple-to-Use Menu-driven MaxPro attachment software. Other data keys may be patched out in the software rendering it totally vulnerable. MAXPRO protected files cannot be patched. Choose whether or not to shut down the program if tampering occurs.

Optional limited life and Usage information collection facilities are provided.

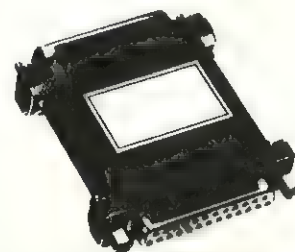
No source or object code modules needed.

The MaxPro system works on IBM; PC; XT; AT Compatibles and PS2.

For additional information contact us at:

Brent Communications
Unit 2
Dragon Industrial Estate
Harrogate HG1 5DN

Tel: (0423) 566972
Fax: (0423) 501442



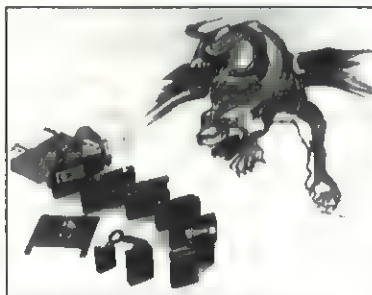
CIRCLE NO. 104

PROTECT YOUR PROGRAM FROM PREDATORS

- Established for over eight years in software security products
- Internationally established product
- Competitively priced keys (from £18.00)
- Supports IBM PC/XT/AT and PS/2
- Supports Apple Macintosh range
- Advanced manufacturing techniques
- Transparent and user friendly
- No drivers necessary (with exception of programmable key)
- Large range of security keys.

For further information and 60 day evaluation kit, contact:

COMSEC (U.K.) LTD.
Cranfield Airport,
Cranfield,
Bedfordshire,
MK43 0AL
Tel: (0234) 751203
FAX: (0234) 751082



COMSEC
(U.K.) LIMITED
COMPUTER &
COMMUNICATIONS
SECURITY SYSTEMS

CIRCLE NO. 105

System Architect™

**A PC based
Microsoft
Windows CASE
tool**

Microsoft Windows
V 3.0 compatible



118-120 Warwick Street,
Leamington Spa, Warks
CV32 4QY

TEL: 0926 450858

CIRCLE NO. 107

8051 C COMPILERS - THE KEIL ALTERNATIVE...

KEIL C51-At Last, A Practicable C Compiler For The 8051 Family
True ANSI C compiler written specifically for the 8051, not a modified generic compiler. Allows C programmers to get 8051 applications up and running fast using many 8051 specific extensions. Gives Intel OMF51 and expanded OMF51 object files for debugging with in-circuit emulators or DScope51 simulator. C51 includes linker, librarian, 6 runtime floating point/integer libraries.

DScope51-8051 CPU Simulator And High-Level Debugger
Loads object files, high-level language lines and symbols from C51, A51, ASM51 and PLM/51. Offers breakpoints, watchpoints, disassembler, display/modify and tracing via an emulator-like user interface. Hardware interface available.

C51 Development Kit-Complete 8051 Tools Set
Includes C51 compiler, A51 Assembler, DScope51 simulator, MONITOR51 hardware monitor and EDIT text editor. This kit allows program generation, debugging and Eprom blowing (via an Eprom programmer) at a very low cost.

teletest51 high-level debug in-circuit emulators available for hire/purchase to cover critical phases of 8051 project development.

Telephone us now for a free demonstration disk and information pack!

Hitex (UK) Ltd,
European Microdevelopment Specialists,
Sir William Lyons Road,
Warwick University Science Park,
Coventry CV4 7EZ.
Tel:(0203) 692066. Fax:(0203)692131.

CIRCLE NO. 119

DISK CONVERSIONS

DISK/DISK • DISK/TAPE • TAPE/TAPE

We can transfer your data across more than 1,000,000 permutations of Micros, Minis, Mainframes, Typesetters, and WPs. We are now able to produce full layout translation on many WP formats. Specialist programming available for complex data conversions.

TAPE

1/4" - QIC-11, QIC-24, QIC-150, (QIC-02)

1/2" - 800, 1600, 3200, 6250 BPI

ANSI, CPIO, DEC, 'DOS' IBM Labelled / Unlabelled / ASCII / EBCDIC, Binary, ICL 6 bit, SYTOS, TAR, VAX Backup / Copy, VME, XEROX LASER.

DEC- TK50, TS11, TU58

Disk Packs - RLO2, TANDON Ad-PAC

FLOPPY DISKS

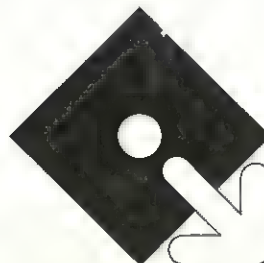
8", 5 1/4", 3 1/2", 3" - Acorn, Amiga, Amstrad, Apple, Archimedes, Atari, CP/M, CP/M86, Displaywriter, FLEX, IBM, Idris, ISIS, Linotype, Macintosh, MDOS, MS-Dos, Olivetti, OS/9, PC-Dos, RT11, Torch, Unix, Victor-Sirius, Wordplex, Xenix, and 1000's more.

DUPLICATION

Most 5 1/4", 3 1/2", 3" and tape formats - please phone for costs

A.I. GROUP SERVICES

Dataprep; Direct Mail; Mailsort (PO approved); Label-printing; Keyboarding; Programming; Kermit Distribution; Specialist hardware; Design; OCR/Scanning; DTP/Typesetting; Printing/Binding/Finishing; PostScript Linotronic bureau; PostScript Fonts; Software Packaging.



Ring or write for full details usual terms C.W.O.

A.I. DOWNLOADING SERVICES

VOYSEY HOUSE
BARLEY MOW PASSAGE
LONDON W4 4PT
FAX 081-994 4959
081-994 5471

A10

CIRCLE NO. 10

BEST BUY

"So simple to use you wonder why other packages make such a meal of resource allocation"



"InstaPlan offers the features of a £2,000 package with none of the trade offs of its low cost competitors."



"We would recommend most warmly... InstaPlan as the best full featured project management package."

Practical Computing

(compared with PMW, Super Project and Harvard).

"InstaPlan is excellent, much better value than PMW, but keep improving it" Mr D Symington, CS Consultants

NEW InstaPlan 5000

InstaPlan 5000 is the latest addition to the InstaPlan family of project management software, which already has 25,000 users, from large multinationals to one man businesses, and is fully compatible with the rest of the InstaPlan family.

Easier Departmental Management

InstaPlan 5000 can gather individual project plans and consolidate them for departmental review, all with a single command on a stand alone machine or a PC network.

By bringing together the department's schedules and resource requirements in one model, you can recognise and resolve staffing conflicts and project priorities long before they become crises.

Most project management systems require all projects to be updated to the same date before merging. This is often impossible. InstaPlan 5000 allows projects to be combined freely.

Planning and Control for Larger Projects

InstaPlan 5000 will run on a standard 640K PC. Just like the rest of the InstaPlan family. But InstaPlan 5000 also supports EMS (Expanded Memory). This means that you can handle in excess of 10,000 Activities. Within these limits you are only constrained by the amount of EMS memory you have available. This is many times the capacity of some of InstaPlan's more expensive competitors.

Better Resource Management

You can now level the load on a group of resources simultaneously, using the Adjust Loading command. The Locate Overload facility will scan all future periods for a resource, identifying any overloads.

We believe InstaPlan 5000 will provide the best solution for professional project managers and busy line managers who need project management to enhance their own and their department's performance.

See what InstaPlan users say:-

Easy to learn and Use

... it's design is just right. I set up a workload plan without even using the manual or any previous planning experience.

David Allen Computer Services Manager
Vallectric Ltd Electrical Contractors

and for Project Managers

I am surprised to find myself learning from this product - an excellent investment.

Mr P M Cleary Project Manager
Maritime Dynamics Ltd

compared with the competition

Having suffered with Project Manager Workbench and SuperProject Expert, InstaPlan has provided us with the tools to produce all our reports and regular project updates easily. Management are impressed by the ease with which they can read and understand the reports.

Ms Petra Meyer Product Manager
Traucom Ltd

In summary

InstaPlan contains all the features and facilities required for project tasking, costing and tracking. An excellent, good value package, it forms the basis of successful project management within a business environment.

Andy Coster QA Manager
Logsys



With Quality Documentation

... a very good package, with excellent documentation.

Mr J Clarke Finance Director
Ceraf Ltd Building Contractors

... And Support too

... Good telephone support

Dr Robin Wilson
Harrogate Borough Council

Providing the Right Reports

... provides me with succinct and easy to understand documentation for Board presentations.

Dr R Gailey Director of Research
Tootal Thread Ltd

For Line Managers

Even though our busy line managers are only occasional users, InstaPlan gives them the tools to plan quickly and effectively.

Brian Alldridge MIS Manager
Prestwich Holdings plc

Money Back Guarantee

We are confident InstaPlan 5000 is the easiest project management system to learn and use, offering a significant price/performance advantage over all other systems. Buy InstaPlan 5000 now, if you return it within 30 days for whatever reason, we guarantee to refund your money by return.

DEEPAK SAREEN ASSOCIATES

Freepost, 28 Preston Rd, Wembley HA9 8BR
Phone 01 908 2423 Fax 01 904 6668

InstaPlan 5000 £499 plus £6 p&p and VAT

☐ Please send me InstaPlan 5000.

☐ I enclose a cheque for £580.75

☐ Please bill my Visa/Access Card

☐ send me more details of InstaPlan 5000 and the InstaPlan family, from only £99.

Company

Name

Signature

Phone no.

Please enclose a compliments slip showing your required delivery address and credit card details if appropriate.

EXE 6/90

Configuration IBM XT AT PS.2 or compatible running under PC MS-DOS 2.1 or later with at least 500K of DOS memory and 3M of hard disk available to InstaPlan. Plan storage requires further DOS memory or access to EMS memory. EMS memory (LIM3 2+) supported to a maximum of 8Meg. A 5000 activity plan will require at least 2 Meg of expanded memory. InstaPlan 5000 will therefore run across networks using network shells like Novell where there is sufficient DOS memory available to InstaPlan. Printers IBM ProPrinter & Graphics Printer Epson FX1 X1 LQ Toshiba 351 HP LaserJet PLUS Series II Deskjet. Input via keyboard ASCII. Think Tank or READY! Output to ASCII, Lotus or dBase. Charts are stored as bit mapped graphics and can be printed but not viewed on screen. ALL MARKS ACKNOWLEDGED. DEALER ENQUIRIES WELCOME. QUANTITY DISCOUNTS AND SITE LICENCES AVAILABLE. TRAINING AND CONSULTANCY AVAILABLE IF REQUIRED.

.EXE APPOINTMENTS

Group Recruitment Sales Manager
Jane Lyons 081-994 6477 x2238
Recruitment Sales Executive
Jeremy Cox 081-994 6477 x2337

Rates: 1/4 page 132mm x 88mm £520
1/2 page 132mm x 179mm £955
1 page 273mm x 179mm £1795

Copy Deadline: 15th of the month of publication.

Process Communications Ltd, 10 Barley Mow Passage,
Chiswick, London W4 4PH Fax No: 081-994 1533

Cancellation: Advertisements may not be cancelled after
the copy deadline.



ixsys

THE LEADING PLAYER IN THE COMMS MARKETPLACE

Allow us to fully exploit your talents and experience in:

DATACOMMS/NETWORKING
TELECOMMS HARDWARE AND SOFTWARE
REALTIME SOFTWARE

and provide you with all the best options for your next career move

Contact Julie Poole on 0442 231691 office hours or 0442 69740
eves/wkends. Alternatively mail CV to Ixsys, Hempstead House,
Selden Hill, Hemel Hempstead, Herts, HP2 4LT.

ETHICAL RECRUITMENT & CONSULTANCY

We are specialists in applying ethical criteria to all areas of business, particularly in technology.

Our emphasis on SOCIALLY USEFUL TECHNOLOGY and our EQUAL OPPORTUNITIES approach produce
a resource of exceptionally motivated and socially aware people, at all levels and in all disciplines.

Whether or not you wish to enhance your career by getting more of the same plus something extra, or are
looking for a new angle on life - we will listen to you, and then to our client organisations and with your
good will, make the match. More than occasionally inspiration occurs.

TRY US

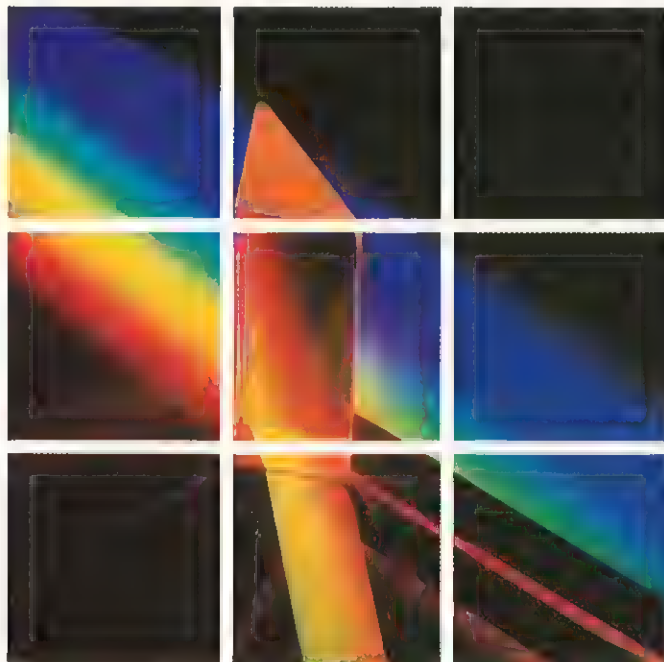
E-mail
(GreenNet)gn: exresource
GeoNet)geo2: exresource
Fax 0225 469673

EXCHANGE RESOURCES

Ethical work in technology

28 Milsom Street
Bath BA1 1DP
Telephone: 0225 469671

SHOW YOUR TRUE COLOURS



REAL TIME SOFTWARE DESIGN - C, UNIX, 68000

£16-24,000

Hertfordshire

To make the most of your potential you need the right environment. The best resources, an atmosphere that stimulates creativity, and the flexibility of a truly multidisciplinary approach. Factors that have led to Crosfield's development of the world's most advanced colour scanning, studio, and data transmission systems for the graphic arts and pre-press industries.

To build still further on our leading status, we have invested substantially in the latest CASE, simulation, CAE tools and SUN workstations.

Involvement in a wide variety of projects mean your expertise will be broadening constantly. Your career path could take you towards project management or technical consultancy. It's up to you.

DEVELOPMENT ENGINEERS

With at least 3/4 years experience of software design and implementation, you'll make a valuable contribution to the development and release of new systems. Your expertise should include exposure to some of the following:

- C, UNIX, X-windows
- 68000 family and RISC processors
- Network Protocols
- Network Management

Knowledge of image processing would be an advantage.

SUPPORT ENGINEERS

Again, we're looking for experienced professionals with the skills to provide invaluable support to existing systems, or those looking for a career change to software support. Knowledge of the following would be a major advantage:

- Assembler
- Macro-II
- 6800 and 68000 assembler
- C and UNIX

In addition to highly competitive salaries we offer benefits including profit share and, where appropriate, relocation assistance. Contact Tony Culpin on (0442) 230000 for an informal discussion, or leave a message on our 24 hour answerphone: (0442) 42557. Alternatively fax your CV or write for an application form to Crosfield Electronics Limited, Three Cherry Trees Lane, Hemel Hempstead, Herts HP2 7RH. Fax (0442) 232301.



A Du Pont & Fujifilm Electronic Imaging Company

PEOPLE IN UNIX

NORTH OF WATFORD SPEAK TO ACTIS FIRST

Do you want to reach the top of the ladder? If you are young, ambitious and self motivated, and have gained some experience, probably but not necessarily, in a UNIX environment and now want to progress further speak to us. We have opportunities for programmers, software engineers, trainers, technical and support staff. You should have experience in any of the following:

C Programming
INGRES/ORACLE/UNIFY/SYBASE
INFORMIX (4GL and SQL)
UNIX, OS9, X.25
COBOL, MODULA 2
PLCs, ASSEMBLERS
PROCESS CONTROL

Telephone HONOR LINDSEY on 0204 20200
or send your CV to ACTIS RECRUITMENT
17 CHORLEY NEW ROAD, BOLTON BL1 4QR

*Unix is a trademark of AT&T

ACTIS
RECRUITMENT

SONIC

RECRUITMENT SERVICES

ANALYST PROGRAMMERS

LONDON - SALARIES TO £25K

'C' Programming skills in a UNIX environment are required by a major international information systems company for the development of commercial applications.

PROGRAMMER/SOFTWARE ENGINEER

WEST COUNTRY - SALARIES TO £25K

Leading Network Systems company in attractive location wishes to strengthen its development teams with 'C' and UNIX professionals. Telecomms experience an advantage.

TECHNICAL PROGRAMMER

BUCKS - SALARIES TO £33K

Financial systems software house seeks to expand its systems programming team. Top class role with progressive company. Technical Skills: 'C' (Essential), LAN, MS-DOS, OS/2.

Contact Sonic Recruitment on
071-734 4157 or 071-722 9268 (eves.)
to discuss these and other opportunities.

Premier House, 77 Oxford St, London W1R 1RB

Analyst/Programmer to £20k + benefits

Established & prestigiously located Human Resources Management Consultancy requires Analyst/Programmer to work on Networked PC's and AS/400. Systems design and integration experience is essential, as is relational database and some 4GL knowledge. AS/400 experience, while preferred, is not essential, and this is an ideal opportunity to expand such skills.

COBOL Programmers - Any Hardware to £21k + benefits

Major Dealer and Software House requires COBOL programmers for retraining to INGRES. UNIX and/or some 4GL experience would be preferred, although commercial COBOL programming is the only essential as full training will be provided.

RPG III/COBOL Programmers - AS/400 to £26k

City-based financial information company requires AS/400 Analyst/programmers, preferably with RPG III on AS/400, although cross training will be provided from System 38. Also of interest is COBOL programming experience on S/3X or AS/400. In all positions a minimum of 18 months commercial programming experience is required, possibly within a financial environment.

C Programmers to £20k + benefits

INCLUDE OPPORTUNITIES.h /* THE BEST AROUND */
INCLUDE EXPERIENCE.h /* 2 YRS + REQUIRED */
DEFINE ENVIRONMENT "Development"
DEFINE PACKAGE "Excellent"
main ()
printf ("call now or send cv for immediate interview/n")
printf ("we meet all relevant candidates/n evenings also/n")

RONICOM RECRUITMENT - A service to suit all candidates

- * Genuine job vacancies at all levels
- * Only appropriate interviews arranged
- * Absolutely no mailshots
- * Comprehensive company information supplied
- * Constructive interview technique advice
- * Experienced and technically aware consultants

EMPLOYERS - Don't Despair!

Ronicom Recruitment, a specialist computer recruitment consultancy, will ensure that:

- * Only curricula vitae appropriate to your requirements will be forwarded
- * All details of applicants' professional and monetary expectations will be provided
- * Professional advertising advice and assistance will be given freely and without obligation



46-47 Pall Mall, London SW1Y 5JG

Telephone: 071-321 0245 Fax: 071-839 7629

SENIOR PROGRAMMER

(Communications) London WC1

This opportunity will enable you to utilise your programming skills in a major initiative to enhance the lives of deaf people. You will join a small team developing a text telephone relay service to enable deaf people and hearing people to communicate over the telephone network.

The person appointed will have experience in programming in C in the PC environment and in one or more of the following areas:

- * serial communications (80x86 assembler and interrupt handlers)
- * porting data to/from database and financial applications
- * PC networks

You will be developing a series of programmes to provide the most effective service to the subscribers and a knowledge of the principles of telephony or data communications would be a distinct advantage. The post is based in London although some travel to the telecoms centre, which is likely to be in a provincial city, will be necessary.

This is a short term post and will suit someone who is looking for a challenging job for the next 12 months. Salary negotiable to reflect experience.

Closing date: 18th July 1990.

For further information and an application form, contact the Personnel Department, RNID, 105 Gower Street, London WC1E 6AH. Telephone: 07 383 3152 (answerphone) or contact Mike Spanner or Brian Glover for an informal discussion on 071 387 8033.

The RNID is working towards equal opportunities and particularly welcomes applications from deaf and hard of hearing people.

**The Royal National
Institute for the Deaf**



CAD CAM
CONCEPTS

An established Cambridge Company with products in GIS, digital mapping, terrain modelling and image processing is expanding its team of

SOFTWARE AND SYSTEMS SPECIALISTS

working on innovative technical developments of

INTERACTIVE 2D AND 3D GRAPHICS AND DATABASE APPLICATIONS.

If you are:

- * a C programmer with experience of system specification and design in a scientific or engineering environment or
- * a systems analyst with a good background in database applications development using SQL and Fortran or C or
- * a Unix expert with an interest in software development environments and configuration management

we would be interested to hear from you. Salaries are to £21,000 and career prospects are good. Relocation expenses are available where appropriate. If you would like to know more, please contact Sheila Fenlon or Michael von Rimscha,
CAD CAM CONCEPTS (Recruitment),
Campbell House, Campbell St, Cambridge
CB1 3NE. Tel: 0223 412965 (day/evening)

'C'ITY

analyst programmers

SALARY TO £28,000 pa

We seek candidates with a minimum of 12 months' programming experience gained in either a DEC VAX, Mini or PC environment. Exposure to structured methods would be an advantage as would experience of any of the following:- windows, work stations or relational databases.

The positions on offer involve a high degree of user liaison and analysis, therefore sound communication skills are essential.

RJB act for a number of leading financial institutions which are recruiting experienced 'C' Analyst Programmers to work on a range of applications in Front Office and Accounting systems

For more information or to arrange an immediate interview please telephone Craig Robertson or Jan Kirkpatrick on 071-436 3701, fax or post your CV quoting reference JK177.

recruitment
systems
RJB



RJB Manpower Ltd,
16/17 Little Portland Street, London WIN 5DE.
Tel: 071-436 3701. Fax: 071-436 8936.

WEST YORKSHIRE

Programmers (COBOL, 'C', BASIC and 4GLs), Analyst Programmers, Software Engineers, Project Leaders, Support (pre and post sales), Trainers (MSDOS, UNIX and Applications).

FOR YOUR NEXT CAREER MOVE
AROUND WEST YORKSHIRE:

Telephone Vincent Atherton
on Leeds (0532) 504560
or write to:

AIREDALE RECRUITMENT
Realtex House, Micklefield Lane,
Rawdon, Leeds LS19 6AX.

**AIREDALE
RECRUITMENT**

SOFTWARE - C/UNIX UK TO 25K

SIMULATION/MODELLING

Sussex £18,000 Intermediate and senior software engineers required, ideally with experience in Real-Time Simulation using C and Assembler, to work on advanced simulators and training systems.

SENIOR SOFTWARE ENGINEERS

Berks £19,000 Degree in Electronics, Computer Studies, Physics or related discipline with 2 years experience of embedded software design. A knowledge of high-level languages such as ADA, Pascal or C needed for design and implementation of Communications applications.

TRANSPORT SYSTEMS

South Coast £20,000 Major development of a motorway control and communication system using embedded software, utilising structured design methods, RT-SASD and TEAMWORK with INTEL 8086 and C.

SOFTWARE TOOLS DEVELOPMENT

£Lots Small team environment, developing software tools for performance analysis of cellular networks. To succeed you will need at least 2 years Real-Time experience and UNIX, C, Sun, DBMS or other software tools.

I.T. INDUSTRIAL SYSTEMS

Hants £12 - £30,000 Our client, a leading software organisation has openings for software engineers to work in the oil and gas utilities market place using UNIX, C, ORACLE, YOURDON etc. Great scope for advancement.

MANY OTHERS

'C', ORACLE, C++, INGRES, UNIX, 4GLs, PASCAL, ADA, INFORMIX & SSADM

STS Recruitment have over 10 years experience in recruitment so to secure your next career move, contact us today. Send a detailed CV in complete confidence or telephone for further information quoting ref. XE/220



33 Staple Gardens,
Winchester,
Hants SO23 8SR

Tel: 0962 69478
Eve/Weekends: 0962 760241
Fax: 0962 841982

DEVELOPERS' MARKET HOLDS FIRM AS INTEREST RATES BITE

The overall rate of recruitment across the IT sector is slowing down for the first time, but technical staff with practical 'hands-on' development skills are expected to enjoy a fairly steady market.

Research by the 60,000 member Computing Services Association shows that there was a drop over the last year in both the growth rate of the industry and the take-up of new staff.

Tony Lewis, Membership Director of the CSA, said that between 1984 and 1988 the annual growth rate in the number of employees was just over 11%. Last year it

rose to 14%, but this year he expects it might drop back to 11% or lower.

Similarly, the growth rate measured in turnover has topped 20% over the last five years, but may fall below that figure in 1990.

Lewis, however, remains optimistic: 'Some people look at the situation and say it's a disaster, but it certainly isn't. 73% of members said business was better than last year: it's still growing, but it's the rate of growth that's slowing. It's a sign of a maturing market.'

David Bevan, Senior Consultant at one of the largest recruitment agencies, Computer

People, said the PC market remained strong and was largely sheltered from fluctuations in the business economy: 'Demand is increasing, and the number of people with PC skills is growing with it, so it remains healthy.' Because supply currently matches demand, salaries are also staying stable.

Bevan added that the areas of greatest current demand include PC technical support, and quality assurance throughout the systems and programming sectors. Firms are also increasingly likely to demand business skills on top of technical training.

PAYING THE PRICE TO BUILD A GREENER CAREER

Computer staff are choosing to take salary drops of up to £5000 to avoid working for firms with 'unethical' business interests.

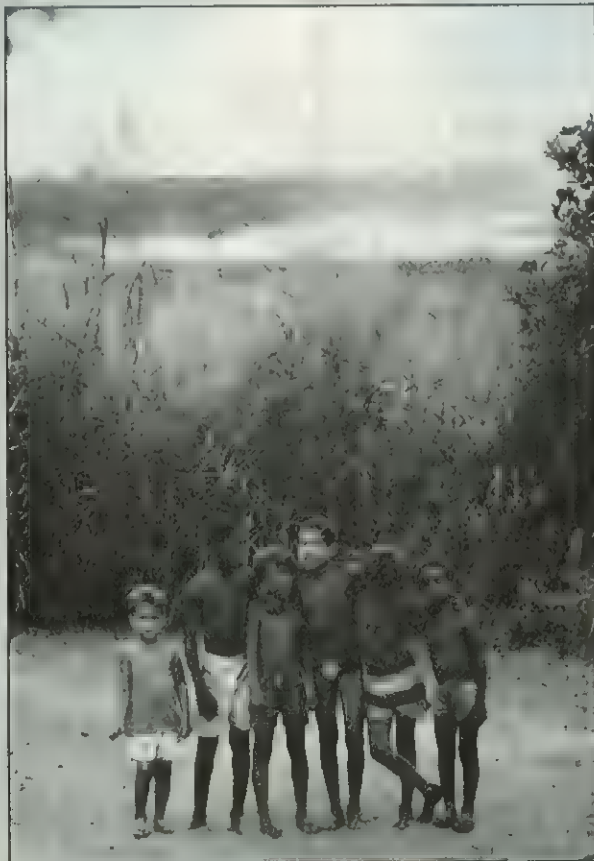
Companies working in the defence or nuclear sectors, or with interests in white South Africa, are being vetted through an ethical recruitment agency, Exchange Resources, based in Bath. Set up four years ago by a former contractor who used to work in nuclear weapons development, the agency attempts to place staff in companies that meet certain principled criteria and which offer equal opportunities.

Some of the largest computer suppliers have appeared on the company's blacklist, although there is some flexibility and, the company admits, some inconsistency. Candidates have been offered positions in particular divisions of offending multinationals if the branch itself passes the test.

Tony Wilson, Founder and Managing Director of Exchange Resources, said: 'We try to encourage individuals to work out what they want from their working lives, and to go for it.'

He added: 'It's amazing how many people we get who are prepared to drop £1000, £2000 or £5000. As people eventually realise that the world is limited and economies can't go on to grow at the rate they are now, they're happy to settle for less.'

This month, Exchange Resources will open two franchises. Wilson, who claims that the managing director of one multinational computer supplier had told the firm 'never to darken their doorstep again', also intends to develop a formal vetting procedure to replace what is currently a fairly ad hoc system. At the moment, the suitability of potential employers is determined from telephone calls, company reports, visits, and contact with groups like the Anti-Apartheid Movement.



Would you lose £5k to stop the Amazon burning?

SUSAN CUNNINGHAM

LONG WAIT FOR GUIDANCE IN A HAZARDOUS MARKET

Applicants worried about the quality of training courses will have to wait at least another 18 months before a government-sponsored accreditation scheme becomes operational.

The IT Industry Lead Body, a working group set up by the National Council for Vocational Qualifications, is defining a series of standards, or 'competences', which training organisations will need to prove they can meet. Set up two and a half years ago, the Lead Body does not expect to have developed a comprehensive package before early 1992.

In the meantime, experts warn that IT specialists are in danger of falling prey to the occasional 'cowboy' operator. Rob Roseveare, Careers Division Manager at the National Computing Centre, said the issue of the standard of training was becoming acute. 'People don't perceive training as a level of investment in staff and are likely to take the cheapest solution. The standard outside this country is not all it should be: inside the country, there are more good options around, but there are courses which are oversold.'

He added that potential candidates should look at what criteria are being applied for acceptance on the course - does it include psychometric tests, or is it based on the size of your cheque book? Not surprisingly, he also recommends that candidates look favourably at established course providers.

Barney Gibbons, Chairman of the Lead Body, said the working party had been 'sweating away' for two and a half years. It should have completed 80% of the competences by the end of this year, although the remaining 20% would be the toughest to define. As a stopgap, the NCVQ intends to offer conditional accreditation to individual courses for two years.

Keith Rogers.

CAREER DEVELOPERS

SOFTWARE MANAGER

SURREY

to £25k + CAR

My client, a small but well-established & expanding Systems Design Consultancy with excellent reputation in the International Aerospace Industry seeks Senior Software Engineer to head up new Software Dept.

Ideally in your mid-late 20's, with varied low & high level languages plus experience of Structured Techniques (eg, ADA, JSP etc) & the ability to present solutions to Junior Engineers & Co. Directors alike. Tremendous career opportunity affording continual technical variety, UK & International travel - and the chance of playing a key role in shaping the Company's Software Development strategy.

C + ADA = £££'s!

GUILDFORD

to £25k pa

Very prestige Software Engineering Consultancy, providing specialist Development Services to the UK Aerospace, Energy & Control Systems Industries NOW recruiting Software Engineer at Project Manager & Consultant levels.

Ideally you will have good Hons degree in Comp. Science or Maths + 3 years "hands-on" in Real Time Digital Systems to include 2 Assemblers, & 'C' in an ADA (or similar) structured environment. Very Professional Company, excellent working environment, committed & enthusiastic Management Team - superb career opportunity for the forward looking young Professional.

Contact David Bird on

081-549 6441.

59 Eden Street, Kingston, Surrey KT1 1BW.
Tel: 081-549 6441. Fax: 081-549 6462.



HIGH TECHNOLOGY RECRUITMENT

Executive

The Azure Project (15K to 25K)

Estate Computer Systems are building a team of specialists for the initial & continuing development of high performance database software. ECS have a strong background in this area and have become a market leader using their "TheDataMachine" product in Property Industry Software. The Azure Project Team will create new software written in C under OS/2, UNIX & DOS to offer total integration with industry standard packages such as WordPerfect working to the high standards required for the 1990's.

We are recruiting specialists for the major area of the project. Each member of the team will have a C programming background and a theoretical as well as practical understanding of computers.

- **Database Specialist** Creating the database storage & access routines providing fast, low level routines to enable easy, high speed use of the system.

- **Printer Specialist** Producing routines to take complex documents created by the main database software and/or WordPerfect and printing them to DeskTop publishing standards.

- **Presentation/Graphics Specialist** expert in all areas ranging from scanner input to creation of graphics and on to a Graphical User Interface.

- **Language Specialist** Formal understanding of computer languages will be essential to work on two language projects: the automatic conversion of C source between OS/2, UNIX and DOS and the compilation of The Azure Project's database language.

We are looking for individualist in a team which will be based in Sleaford in Lincolnshire; for programmers who love computing and who want the opportunity to create a world class product in a progressive and expanding company; rewards for the successful applicants will be substantial with initial salaries starting at £15,000 to £25,000 depending on experience and background. Write (not phone) with a full CV to Barbara Westrup at our Sleaford office for more details of this exciting opportunity.



OVER SIXTY YEARS OF PROPERTY EXPERIENCE

Allen-Miles

TECHNICAL RECRUITMENT

SOFTWARE ENGINEERING Opportunities £11000 to £25000

We have a number of clients throughout the UK with urgent requirements for Software Development and Support Engineers. The following are a very small selection.

SOFTWARE ENGINEERS DORSET TO 15K
IBM P.C.'s, MS-DOS, BIOS, 8086/80286 Assemblers and 'C'.
Graphics involvement and H/W Interfacing.

SOFTWARE ENGINEERS HOME COUNTIES 11K TO 18K
ISDN development. Experience of Comms based Data Interfaces using 'C' UNIX/PASCAL.

SOFTWARE ENGINEERS HERTS c20K
Development of Integrated Office Systems. Experience in the I.T. Industry, using 'C' UNIX/DOS.

SOFTWARE Q.A. MANAGER HANTS TO 25K
Organisation / Management of an S.Q.A. Group, providing cost effective Q.A. for Software / Firmware design/development.

SENIOR SOFTWARE ENGINEERS HOME COUNTIES TO c18K
Real-Time Micro based S/W development and test. 'C'/Assembler/UNIX/MS-DOS. Logic Analysis/ICE.

For further information, please phone **ALAN THAKE** or **IAN COLLINS**. Alternatively send your C.V. to the address below.

111a Seabourne Road ● Southbourne ●
Bournemouth ● Dorset ● BH5 2HF

TEL: (0202) 417750

to 20K plus benefits

MS-DOS Programmers

We are a rapidly expanding software house, specialising in developing and marketing of communications and office automation products for major UK organisations, including British Telecom.

We're looking for an energetic programmer to join our committed and professional team to work on a major new exciting project.

Interested? The skills you'll need to apply are:

- software design
- minimum of 2 years 'C' programming
- developing applications to run under MS-DOS
- a good communicator.

It would also be an asset if you've been involved with TSR and device driver programming, software engineering of comms or office automation products and are familiar with 8086 Assembler.

The rewards? We're offering an attractive package including a salary of up to 20K, a company bonus scheme, BUPA and a strong commitment to advanced training.

If you think you're the person we are looking for write to:

The Technical Director
The PMS Group
Norfolk House
Smallbrook Queensway
Birmingham B5 4LJ

DIP Ltd has designed, developed and produced THE IBM-compatible POCKET PC, sold as PORTFOLIO. They are currently writing an enhanced PC compatible BIOS for palmtop computers. Development will be exciting and fast-paced, pushing the IBM PC architecture to its limits.

**SOFTWARE ENGINEER,
GUILDFORD, to £18,000**

IBM PC BIOS + DEVICE DRIVERS at DIP

Ref: 132569/jh

If you are enthusiastic, know 8086 assembler, DOS and are interested in a career with this young vibrant company, please contact their consultant Julianne Hull.

TECHNICAL PERSONNEL

York House
Chertsey Street
GUILDFORD
Surrey GU1 4ET
Telephone: 0483 65566
(24 hours)



UNIX PRE-POST SALES

**London, Home Counties
to £25,000 + Car**

If you are currently working in a UNIX support role and are looking for the rewards you know you truly deserve then look no further. This leading manufacturer and systems house is looking for UNIX professionals with expertise in one of the following areas:

OFFICE AUTOMATION
COMMUNICATIONS
4GLs-RELATIONAL DATABASES

Excellent communication and presentation skills are essential. In return for your expertise you will be joining an elite team of UNIX professionals in a company that has been growing at 70% p.a. for the past 2 years.

Please ring our selection consultant JERRY HERD for an informal discussion on 081-504 2204 (office). Or alternatively send a detailed CV (including salary and preferred locations) to Michael Bailey Associates (Recruitment Consultants), Rococo House, 281 City Road, London EC1V 1LA.

MBA

Michael Bailey
Associates
Recruitment Consultants

MBA

SKILLED IN UNIX and 'C'??? MOVE INTO SUPERCOMPUTING!!

Contrary to popular belief you don't need experience of Occam programming, transputers or the like to move your career into the expanding market of superperformance computing. What's more you can choose between support or development. Take a look!!

UNIX-based Software developers, To £24000 plus Car!
West London. Working in 'C' at system level to port DBMS, Graphics, Risk Management and Decision Support software onto massively parallel architecture!!

UNIX and COMMS Support, To £25000 plus car!
Thames Valley. Join the company who dominate supercomputing! Second level support! Pioneering OSI software! Large Systems background essential!

UNIX Internal Consultant, To £21000, Cambridge.
Expand your skills in UNIX by providing a technical bridge between development and suppliers! Training offered in transputers, Occam, OOP and Windows!

UNIX-based Support, To £30000 plus Car, M3/M4 triangle. Pre-sales support and Systems Integration for the fastest-growing company in supercomputing! Benchmarking 'C' and FORTRAN skills essential!!!

Interested??? If you want to by-pass the dying era of sequential computing and rise to the challenge of the new world of parallelism contact **ALAN CARNELL** at **Concurrent Appointments**, the consultancy with a focus on this market.



**Concurrent
Appointments**
Software
Recruitment

27 FIELD CLOSE
HARPENDEN
HERTS
Tel: 0582 712976
Fax: 0582 764858

THE UNIX RECRUITMENT SHOP



MANAGEMENT CONSULTANTS, Central London (£25k - £Neg)
• 8 + years Development & Management experience
• Experience of OSI Architecture
• Understanding of emerging Open Systems Technologies
• Effective communications with people at all levels
• Understand the latest concepts in Quality and the use of modern design tools

SENIOR SOFTWARE ENGINEER, Portsmouth (£20k - £Neg)
Commercial experience a must.
• Good interpersonal, presentation & project planning skills.
• Experience Informix 4gl essential.
• Office Automation | Networking
• Experience of Datal



EIS/MIS SUPPORT/CONSULTANT, Berks (£20k - £Neg)
• Excellent Communication Skills
• Technical background in IBM, DEC or Unix Environment
• High analytical and problem solving abilities
• Willingness to travel overseas.



PROGRAMMERS, Middlesex (£11k - £16k)
• 3 + years programming experience in C.
• Working in an MS-DOS or Unix Environment
• Applications programming of a commercial nature.
• Experience of EPOS



Open Systems People Ltd is a technical recruitment agency specifically aimed at recruitment of staff at Technical Support, Systems Programmer, Project Leader and Middle Management levels. Our clients are involved in 'State-of-the-art' software engineering, in fields such as Electronic message handling, Document architectures, Relational Databases, LAN/WAN, Graphical User Interfaces, and other emerging technologies. If you have a strong background in Unix then make a move in the right direction and contact Open Systems People, your technical friends in recruitment.



Contact Hanif Noormohamed on
0767 317942 - Daytime or Evening
0767 315795 - Fax
for an informal discussion.
Alternatively fax or write

OPEN SYSTEMS PEOPLE
PO Box 446 Biggleswade Beds SG18 0QZ

Unix is a registered trademark of AT & T

SOFTWARE DEVELOPERS

SURREY

Versatile 'C' programmers to develop real-time, interactive, image processing and graphics systems. Any of the following useful: **UNIX, OS9, Object-oriented Design, IBM PC or Macintosh experience**

to £20k

CAMBS

Developing **Database Management Systems and Program Generators**, you should have 'C' experience either under **MS-DOS, UNIX, VMS or FlexOS, MS Windows, Presentation Manager or X-Windows** knowledge very useful.

£15k - £21k

E.MIDLANDS

Leading supplier of **UNIX** based business systems is seeking 'C' programmers and **Informix 4GL** programmers with a minimum of one years experience. Training provided in **Yourdon** and **CASE** tools.

to c£16k

W.MIDLANDS

Rapidly expanding established Communications software house seeking minimum 2 years experience of 'C'/MS-DOS in a commercial or industrial environment. Any of **80x86 Assembler, TSR or Device driver** experience useful.

£15k - £20k

HANTS

Microprocessor Software developer sought by international communications company. You should have 'C' and **Assembler**, hopefully with experience of either **MS-DOS, UNIX, ICE or Structured Design Methods**.

c£18k

LONDON

Designing Personnel Systems for the international marketplace, our client is keen to recruit 'C' programmers under **MS-DOS**. Any **Networking, 8086 Assembler, or Database Design** experience useful.

£15k - £22k

LONDON/MIDDX

International company seeks Software Engineer with **M68000 Assembler** experience to develop Real-time comms. 2 to 5 years' experience.

c£22k+car

HERTS

Electronic Publishing specialist keen to recruit 'C' Software engineers with experience in a real-time, hopefully structured environment. At least one of the following would be an added advantage: **PC's, C++, OOP, UNIX, TCP/IP, NFS, OSI or Postscript**.

£16k - £23k

ESSEX

Small team, developing **LAN/WAN Communications** interest you? Then you should have good **UNIX/'C'** skills - any of the following would be an added bonus: **x.25, TCP/IP, ISO 7, Windows, Networking or UNIX Operating Systems** experience.

to £17k

AVON

Developing **EDI Comms, Insurance and Financial Systems**, you should be an analyst/programmer with good **UNIX** experience with either 'C' or **FORTRAN**.

to £17k

ITALY

Remote Sensing project requires good **FORTRAN** skills on **VAX/VMS** - any 'C', **Interactive Graphics, PV-WAVE** or **signal processing** experience useful.

£18k - £21k

BEDS/BUCKS

Leading consultancy developing modelling software systems is seeking graduates with sound **FORTRAN** skills combined with one or more of the following: **UNIX/'C', X-Windows, Graphics, Database or Q.A.** Support roles also available.

£15k - £30k

HERTS

Developing applications software for the **Secure Systems, Retail, Process Control, Insurance and Health Care** market, is seeking 'C'/UNIX skills with hopefully **SQL, Oracle or Methods** experience, although training will be provided.

to £17k

MANY MORE INTERESTING POSITIONS THROUGHOUT THE U.K!

Contact us to find out more about these posts or the many others that we have. As specialists in the industry, we can provide you with a professional and confidential service in assisting you with the right career move. Telephone or write to one of our consultants.

081-392 1514

ACUMEN SEARCH & SELECTION INTERNATIONAL
No 7, Tideway Yard, Mortlake High Street, London SW14 8SN
Tel: 081 392 1514 Fax 081 392 1518

Evenings till 9pm 081-878 8206



ACUMEN

SEARCH & SELECTION
INTERNATIONAL

Chess Computer Appointments Ltd

Search, Selection and Advertising Consultants
3rd Floor, 94 Great Portland St. London W1N 5PB
Telephone: 071 323 6880
Fax: 071 636 5245

TECHNICAL CONSULTANT - PRE SALES

£295

Prestigious corporation requires high level consultants with an all round knowledge of the PC market and specialist expertise in either **XWindows, Databases, or networking/communications** to administer consultancy to senior I.T. professionals.

PC ANALYST PROGRAMMER

Financial institution have vacancy for ambitious individual with 'minimum 12 months' experience as PC analyst/programmer. Experience of **Turbo** or **Microsoft 'C'**. Knowledge of **Networks** useful.

£17500 + BENS + MORTGAGE SUB

NETWORKING SUPPORT

£18k + Car

Major distributor requires systems engineers with experience of **Novell P.C. LAN or Torus**.

TECHNICAL SOFTWARE SUPPORT

£10k + Car

Micro Computer manufacturer requires 'C' support specialists with 3 months' experience. Probably from an End-User dealership background you will have a knowledge of a variety of 'off the shelf' packages e.g. **dBase, Lotus, Multimate, Pegasus, Smart, etc.** to provide support to dealers, distributors and VAR's.

FINANCIAL INSTITUTION

AS400/RPG Professionals

Environment: **IBM AS/400/RPG**
Relational Database design
Full PC - Networking and AS/400 connectivity

Objective: Complete support of the money market trading settlement operations, dealer-trader decisions support and interface to corporate accounting systems.

D-BASE/CLIPPER OR FOXBASE ANALYST PROGRAMMER

£18k + BANKING BENS.

Stockbroker has opening for experienced database programmer analyst to develop financial applications for their busy dealing room area very exciting work environment.

UNIX SPECIALIST - SUPPORT AND DEVELOPMENT ALL LEVELS - CALL IMMEDIATELY.

Full listing of vacancies available.

Call 071-323-6880 and ask for full details of vacancies.

FOR FURTHER DETAILS ON OPPORTUNITIES IN
PROGRAMMING/TRAINING/ANALYSIS/SUPPORT/SALES/ENGINEERING

CONTRACTS

C++

URGENT

South East

0707 275751

For the latest contracts
why not page the
ORACLE on 285 for
LWT/Thames areas and
page 287 elsewhere

THE COMAC GROUP PLC

Technical Software Vacancies

WINDOWS/OS2/ PRESENTATION MANAGER

London/Thames Valley £18-28K

A joint venture company between a world leading Information Technology centre and various financial institutions is aiming to increase its workforce by a massive 30%. In order to be part of this exciting development you will be degree qualified, have 2 years experience in structured software design and ideally be versed in communications protocol and interface design. In return for your knowledge of the 'C' language, together with MS WINDOWS or PRESENTATION MANAGER you will be actively involved across the entire software life cycle and be offered an attractive salary, bonus scheme, family medical plan, pension scheme and free foreign language tuition! Ref: MJ25100

RECENT GRADUATES M.Sc GRADUATES

Various Locations to £15K

We have a large number of clients ranging from a major international defence contractors through to small, independent software houses who have an urgent requirement for technically qualified graduates with software experience. If you can offer C/UNIX, ADA, PASCAL, COBOL or any Real Time experience we can offer you a number of interesting opportunities. MJ2557

C → C++

N.H. Counties

to £30K

Would you like to develop your expertise in Graphics, Database, Data Comms, Object Orientated Design, Expert Systems or Distributed Processing? Would you welcome the opportunity to TRAIN in C++? If so, and you can offer a good technical degree and a minimum of 1 years experience in C/UNIX you could be working on a variety of technically challenging projects within a small, rapidly growing Systems House. Along with this golden training opportunity my client offers private medical care, company pension, bonus scheme and employee share options. MJ2563

NETWORKING/COMMS SOFTWARE

London/H. Counties to £30K + Car

SKILLS

- * C/UNIX * VAX/VMS * X25 * X400
- * PC-DOS * IBM * OSI * SNA

OPPORTUNITIES

Consultancy, Finance/Retail/Comms, Design/Development, Commerce/Government

BENEFITS

Excellent Salaries, Company Pension, BUPA, private and medical insurance, generous relocation package, varied, interesting projects and superb career development. MJ2570

SECURE COMPUTING

Hampshire

to £30K + Car

Our client, a joint venture company between one of the UK's leading technical consultancies and the UK's foremost communications company is actively seeking experienced software engineers to lead the way in Secure Computing Systems. With good, sound experience in C/UNIX, 68000 ASSEMBLER, 4GL/SQL, FORMAL METHODS or OPERATING SYSTEMS you can expect to break new ground in the Planning, Design, Implementation and Evaluation of Total Security Systems for the Government, Industry and Commerce sectors. Technically challenging, stimulating work in an advanced technology environment in the industries most exciting growth potential field. A generous salary, relocation, company pension, private medical insurance and BUPA complete the benefits package. MJ2565

For more details on these opportunities call **MIKE JENKINS** on 0442 231691 office hours or 0582 456417 eves/wkends. Alternatively mail CV to Executive Recruitment Services, Hempstead House, Selden Hill, Hemel Hempstead, Herts, HP2 4LT or fax on 0442 230063

ERS

STOB - Letter from Parity

Like Sherlock Holmes himself, Verity is cursed with a smarter, elder sibling. Parity Stob is also a computer programmer. She works for WED (Weinstick Electric Death) Ltd, and sometimes corresponds with her baby sister.

Darling Verity, My transfer *finally* came through last week, so am now working on the SMPTJFALHW-FLAH project (Secret Military Project Teach Johnny Foreigner A Lesson He Won't Forget In A Hurry). Had the option of joining Anti-Personnel Missile Control (office near to the coffee machine, nice new ergonomic terminals, but Team Leader has BO problem) or Devastation Maximisation Unit (office has window with interesting view of main road, and Team Leader has hilarious red-haired toupee). Plumped for Ginger's outfit, as in the event of personal side-by-side supervision at terminal, his company more salubrious.

Can't say which country this stuff is for - Official Secrets, and all that - but, to give you a clue-ette, the Operational Requirement states that our hardware must be capable of being shipped disguised as consignment of amusement arcade machines. The DMU CPU module that we are working on is going to be got up as a Fruity. One of the last jobs, before it goes out, is to get it to flash the message WANT A GOOD

TIME? COME AND PLAY ME! JACK-POTS GALORE! on its status lights, in case of a snap inspection at Customs.

Of course, we wouldn't have to clown about like this if it wasn't for the Cold War thaw. Morale is very low. Each fresh piece of good international news brings a wave of despondency. The dragon i/c stationery (you know, the one that wears a face that she should *leave* in a jar by the door) didn't understand why the VAX laser printer was consuming two toner cartridges a month, so she had the operator list the print queue. There were 27 files pending, all from different accounts, all called 'CV.DOC'.

Another source of gloom is the usual antagonism between permies and subbies. Bostock, Head of Secondary Comms, is a particularly self-satisfied son of Thatcher era. He is quite handsome in a middle-aged, smoothie sort of way (if you don't mind grease with your chips), always wears Old School cuff-links and a smirk, and is given to cornering permies and explaining why we are such fools. I went into his office the

other day, to borrow the Secondary Comms Data Dictionary, and received fully ten minutes of this spiel. He ended by saying: 'No Parity, if you want to make any progress in this life, you've got to keep your hand up to the kirt of Fate.' I thanked him for this pearl of wisdom and left pronto for Ladies to wash my hands. Resolved to wear jeans to work henceforth: don't want to suffer Fate's fate.

Bostock is right about career prospects, though. Just look at the company cars. Ginger has been here seven years, loyal idiot, and all he has to show for it is a Ford Yobbo. Head of project only manages a Vauxhall Salesrep, and the flashest car on site is the Chief Security Guard's Ford Fatman. Which reminds me: must go, need to pick up new id pass. Lost original four days ago, since which time have been presenting bus pass to get in. If the gate guard ever took off his mirrored shades, Parity would be dead meat.

Take care, love + kisses
PARITY

EXE

Opportunities for Software Professionals

SYSTEMS/CASE

★ **FIELD APPLICATIONS ENGINEER**
BERKSHIRE **£20k + Car**
 To provide pre + post sales consultancy + support for CASE tools in a real time embedded systems environment, responsible technical role. Exp. req. in ADA and C plus realtime kernels eg PSOS, VRTX, VERSAdos etc.
 Ref/PRF

★ **SYSTEMS ANALYST**
BERKSHIRE **TO £20K + European Travel**
 At least seven years analysis and COBOL programming experience required to help design and build a new European Integrated package for this International company. Honeywell DPS8 and DM4/TP are advantageous as is the ability to speak any European Language.
 Ref/1670/05

REALTIME

★ **SYSTEMS PROGRAMMER**
BERKSHIRE **TO £17K**
 Create new Run Time Systems using 8086 / 80286/80386 assembler and provide maintenance support to all OEM customers. Additional experience of three of the following is essential: DOS, OS/2, COBOL, C, FLEX, UNIX or VM.
 Ref/1630/05

REALTIME

★ **SENIOR PROGRAMMER**
SURREY **TO £16K**
 Must be able to write small program specifications, have a logical approach to problem solving and have around three years experience in areas such as configuration control, CASE Tool support, coding and Testing ADA or PASCAL/ CORAL 66 and VMS experience is necessary.
 Ref/1504/01

★ **SOFTWARE ENGINEERS**
HAMPSHIRE **TO £24K**
 To design & develop radio trunking systems & special customer projects involving the analysis of requirements, upfront task design, writing software, test & integration and system testing, educated to degree standard plus R/T experience using VAX/VMS.
 Ref/ 1460/00

UNIX ★ UNIX ★ UNIX

★ **MATHEMATICAL MODELLING and SIMULATION** **TO £25K**
 Good GRADUATE in Maths/Physics/OR Computing for Complex Systems Design and Analysis or Discrete Event Simulation. Experience of FORTRAN.
 Ref/ D/388

UNIX

★ **C PROGRAMMERS**
NATIONWIDE **TO £22K**
 We continue to experience a massive shortfall in experienced C Programmers and can offer a variety of interesting and challenging positions. If you can develop, implement and test quality Software in C, under either UNIX or MS-DOS then we can probably find you the job you want.
 Ref/1669/

CONSULTANCY/SUPPORT

★ **POST SALES SUPPORT**
BERKSHIRE **£15-20K+**
 Knowledge of manufacturing industry, in particular production control & inventory management. Software experience of RPG111, IBM AS400, IBM 36/38. Duties will include bug fixing, application development and implementation. Travel possible.
 Ref/1381/05

CONSULTANT

★ **SURREY** **£15-20K**
 At least 10 years experience with a thorough computer science background. Should possess experience of small project management with real time skills using ADA, PASCAL, CORAL 66 or VMS
 Ref /1501/01

CONSULTANCY/SUPPORT

★ **NETWORK APPLICATIONS SUPPORT**
BERKSHIRE **TO 23K + CAR**
 This position involves installation of PC Networking applications and its support, network security and database maintenance. Experience of NOVELL NETWORK, LOTUS, EXCEL and wordPERFECT is required along with some exposure to PROGRESS 4GL
 Ref/ 1671/05

MANAGERIAL

★ **SOFTWARE ENGINEERING TOOLS MANAGER** **c£23K + CAR**
HAMPSHIRE
 Minimum 10 years experience (preferably software house background). Extensive knowledge & use of software engineering tools - automate, excelerator, software through pictures, information engineering workbench, Yourdon workbench.
 Ref/1433/00

MIS/DATABASE SYSTEMS

★ **TECHNICAL CONSULTANTS**
LONDON **TO £14-20K**
 Experience of applications design and development using informix or ORACLE. Cross training can be provided with experience of other open systems rdms. This will be an analyst/programmer role involving systems design and programming, client contact and the use of SSADM. Jackson and Yourdon.
 Ref/ 1005/12

OFFICE AUTOMATION

★ **ANALYST PROGRAMMER**
NORTHAMPTONSHIRE **£10-14K**
 To support existing applications for supplied & bespoke software, analyse business requirements, including the development of bespoke systems, installation of new hardware & software. Knowledge of LANs, spreadsheets, dBase IV
 Ref/1494/55

GRAPHICS/CAD/DSP

★ **CAD DEVELOPMENT ENGINEERS**
BERKSHIRE **£15-19K**
 High level language programmers from a CAD/ CAE or similar background needed to develop suites of CAD tools for ASIC/VLSI state of the art environment. UNIX, C, PASCAL, LISP. Knowledge of industry standard tools an advantage.

★ **DSP ENGINEERS**
SURREY/MIDDLESEX **TO £25K**
 Rapidly expanding systems house applying DSP expertise to high tech industries eg satellite communications. Knowledge of DSP software or hardware essential. Excellent salaries and working environment.
 Ref/ D/BOB

For further information on these or related positions call
 ANITA HARVEY ON (0734) 774234 or (0734) 794278 after 7p.m.
 FAX: (0734) 772773

Or write in confidence to CPS at:-
63 Peach Street
Wokingham
Berkshire RG11 1XP

CIRCLE NO. 109



COMPUTEC PERSONNEL SERVICES

.EXE
Please mark 'Later' if appropriate
Notify 050 SAH for address change
EC DISPLAY 2 WKS LIBRARY, EC
PJ WINFIELD HERB/VEG, EC
DH MITCHELL PLANT HEALTH, EC
D-MARE NEMATOLOGY, EC
EC LIBRARY EC

Available for MS-DOS and Macintosh

Date
Seen
17.7.90

02686872/V5*N2/EC.S/2



Natural selection provides unique
passive protection for the porcupine.

The Activator - Natural Selection For Software Protection



Inventor and entrepreneur
Dick Erett explains how
"The Activator" provides
sane protection for your in-
tellectual property.

"In any industry, just as in nature, the
process of natural selection raises one
solution above another. Natural selection is
the most elegant of engineers.

In the area of software protection The
Block has been selected by the market-
place as the solution that works. Over
500,000 packages are protected by our
device.

For the past 4 years our philosophy has
been; 'You have the right and obligation to
protect your intellectual property.'

A New Ethic For Software Protection

In allowing end-users unlimited copies
of a software package and uninhibited hard
disk and LAN operation, The Block has
created a new ethic for software protec-
tion.



"On-the-fly" programmable memory option now available for
OS/2, MS-DOS and Macintosh

By removing protection from the
magnetic media we remove the constraints
that have plagued legitimate users.

They simply attach our key to the
parallel port and forget it. It is totally
transparent, but the software will not run
without it.

A New Technology For Software Protection

Our newest model, The Activator, builds
on our current patented design, and
establishes an unprecedented class of
software protection.

We have migrated and enhanced the
circuitry of The Block to an ASIC (Appli-
cation-Specific Integrated Circuit)
imbedded in The Activator.

This greatly improves speed and
performance, while reducing overall size.
Data protection can also be provided.

Programmable Option

The Activator allows the software
developer the option to program serial
numbers, versions, or other pertinent data
known only to the developer, into the
circuit, and access it from the program.

Once you program your part of the
chip, even we have no way to access
your information.

The ASIC makes emulation of the device

virtually impossible. It also presents
an astronomical number of access
combinations.

Full 100% Disclosure

Since The Activator is protected by our
patent we fully disclose how it works.
Once you understand it, endless methods
of protection become evident.

Just as no two
snowflakes are the
same, no two im-
plementations of
The Activator
are identical.
And like the
snowflake the
simplicity of
The Activator is its greatest beauty.



We never cramp your programming style
or ingenuity. Make it as simple or compli-
cated as you desire.

Let us help safeguard what's rightfully
yours. Please call today for additional
information or a demo unit. It's only
natural to protect your software."



Software Security UK LTD
21A The Precinct, High Street, Egham
Surrey TW20 9PN
TEL: (0784) 430060
FAX: (0784) 430050

Unlimited Copies • Programmable • No Batteries • Small Size • Fast • Patented • Data Protection